CURVELET TRANSFORM WITH ADAPTIVE TILING

A Thesis
Presented to
The Academic Faculty

by

Hasan Al-Marzouqi

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Electrical and Computer Engineering

Georgia Institute of Technology
December 2014

CURVELET TRANSFORM WITH ADAPTIVE TILING

Approved by:

Professor Ghassan AlRegib, Advisor
School of Electrical and Computer Engineering
Georgia Institute of Technology

Professor James McClellan
School of Electrical and Computer Engineering
Georgia Institute of Technology

Professor Framerz Fekri
School of Electrical and Computer Engineering
Georgia Institute of Technology

Professor Anthony Yezzi
School of Electrical and Computer Engineering
Georgia Institute of Technology

Professor Zhigang Peng
School of Earth and Atmospheric Sciences
Georgia Institute of Technology

Date Approved: 11/12/14

ACKNOWLEDGEMENTS

I would like to express my special appreciation and thanks to my advisor Dr. Ghassan AlRegib, for his continous guidance, support and invaluable advice. No words can express the trmendous impact our interactions had on my development. I would like to thank Dr. James McClellan and Dr. Faramarz Fekri for the time and effort spent reviewing this work. Your feedback, discussions and comments during my stay in the Center for Signal and Image Processing (CSIP) were valuable to me. I would also like to thank Dr. Anthony Yezzi and Dr. Zhigang Peng for reviewing the dissertation.

I'm also grateful to Dr. Tamir Hegazy and Dr. Long Zhiling for their comments and feedback about the content of this dissertation. I would like to thank Can Tamel for providing the dataset used in testing the scale selection algorithm. Special thanks go to Mohammed Abed, Zhen Wang, and all the members of MSL and CeGP laboratories.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

INTRODUCTION

The goal of a transform is to build an efficient representation of signal content. Typically, we would like to concentrate signal energy into a small number of coefficients. Such energy compactors forms the backbone of audio, image, and video coding algorithms (*e.g.* MP3, JPEG, and MPEG). The compact representation is also useful in a variety of data processing tasks including denoising, compressed sensing recovery of data points, and feature selection. The sparse representation of data is induced by using a frequency domain representation such as the Fourier transform or one of its variants (*e.g.* Discrete Cosine Transform). A major drawback of the Fourier transform is that it does not provide correspondence between the frequency content and spatial coordinates. The wavelet transform attempts to remedy this problem by dividing the signal into different frequency bands and analyzing each band locally. Wavelets generate sets of coefficients where each set holds the spatial content relevant to a unique frequency band. Wavelet frequency divisions are mostly non-adaptive and independent from the signal content. In 2D and higher dimensions, wavelet frequency divisions are shaped as squares or rectangles of different sizes. Such divisions limit the directional selectivity of wavelets. The curvelet transform [14, 13, 40] and other directional transforms [31, 26, 35] introduce directional frequency tilings. Figure 1 illustrates curvelets tiling of the frequency domain.

Little attention has been given in the literature to the dependency between signal content and the optimal frequency domain tiling. The number of frequency divisions and their locations are regularly selected without regard to signal activity. In the proposed research, our goal is to address the problem of frequency domain tiling

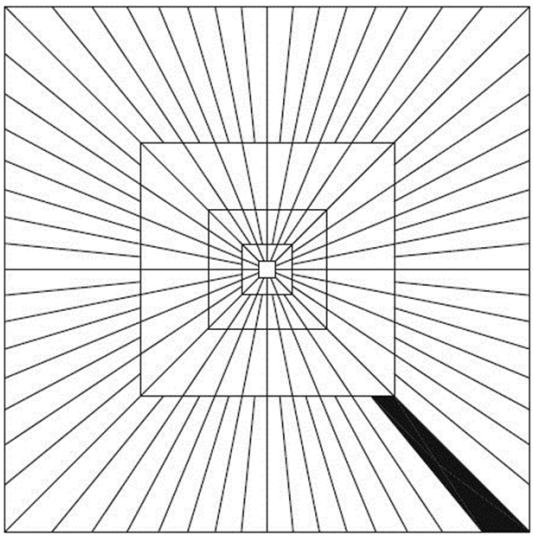


Figure 1: Default curvelet tiling of the frequency domain.

adaptation using the curvelet transform as a basis algorithm. The curvelet transform has shown successful performance improvements in a wide range of application areas including denoising [59, 53, 61], image fusion [17], edge enhancement [46], compressed sensing data recovery[33], deconvolution [28], texture analysis [5], face recognition [44, 66], and content-based image retrieval [60].

A plot of a curvelet coefficient in the spatial domain is shown in Figure 2. The curvelet transform represents images as a combination of such needle-shaped directional elements. Figure 3 presents a sample image and a plot of its default curvelet coefficients. The sample image contains a number of lines at different directions. Curvelet coefficients representing the sample image are shown at three different scales. The inner most level contains a smoothed version of the input image. The second and third levels show line segments included in the sample image, in different curvelet tiles depending on the orientation of the lines. The second curvelet level includes four different wedges per quadrant. The third level includes eight wedges. The empty blocks in the third level indicate curvelet wedges with marginal activity. The third level line segments are smoother and thinner than the line segments in the second level.

Figure 2: A curvelet element in the spatial domain.

In this dissertation, we will adapt the number, size, and locations of curvelet tiles according to denoising performance. The PSNR (Peask Signal to Noise Ratio) of the resultant image is used as the criteria for performance. Besides the clear advantage of developing better denoising algorithms, curvelet tiles trained using denoising provide a better representation of image features. This allows the use of denoising-based tilings in other applications. The method developed in the dissertation introduces a toolkit capable of customizing curvelets to other goals (classification, compressed sensing,...etc.). Naturally, the approach used in this work can be extended to other directional transforms and to the broader family of wavelets and wavelet-like transforms. An alternate optimization cost function that eliminates the training step when dealing with seismic and other unique data sets is also derived in this dissertation.

The developed adaptation procedure is applied to a number of different application areas. Adaptive curvelets are used to solve the problem of sparse data recovery from subsampled measurements. Performance comparison with default curvelets demonstrates the effectiveness of the adaptation scheme. Adaptive curvelets are also used in the development of a novel image similarity index. The developed measure succeeds in retrieving correct matches from a variety of textured materials. Furthermore, an

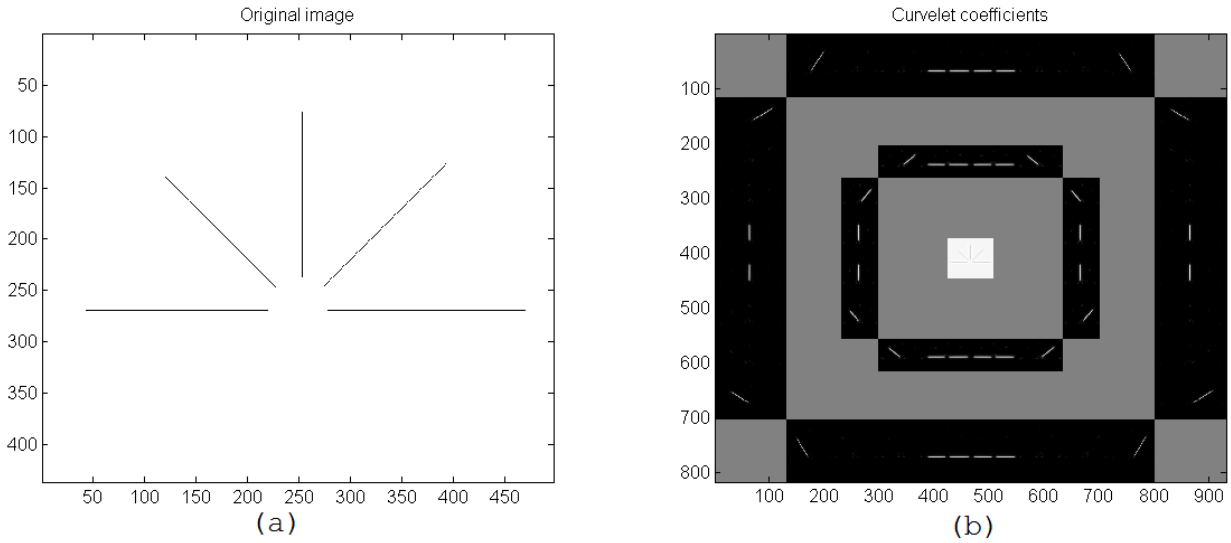


Figure 3: Plot of curvelet coefficients for a sample image [1]. (a) Original image. (b) Plot of its curvelet coefficients. The coefficients are normalized and rescaled for visualization purposes.

algorithm for classifying different types of seismic activities is presented.

The remainder of this dissertation is organized as follows. The next chapter provides an overview of wavelets and directional transforms. The chapter concludes with an overview of the curvelet transform and its technical details. Chapter III presents the developed set of algorithms that are used to search for optimal curvelet tilings. The chapter concludes with experimental results demonstrating the performance of adaptive curvelets in denoising seismic and face images. In Chapter IV, a sparsity-based method for recovery of seismic traces from subsampled data will be presented. The results obtained indicate that for seismic data recovery, adaptive curvelets outperform default curvelets. Furthermore, in Chapter V an image similarity index based on adaptive curvelets is introduced. Adaptive curvelets are shown to succeed in retrieving correct matches from different classes of textured data sets. Examples of these classes include granite, fabric and seismic data sets. An alternate training cost function is presented in Chapter VI, where we show that with certain data sets maximizing the coefficient of variation can be an efficient cost function for use in the

search for the optimal frequency domain tiling. Conclusions and a view of future work are presented in Chapter VII.

CHAPTER II

BACKGROUND

In this chapter, we will present a literature survey of different transforms used in representing localized frequency content of signals and images. The survey includes an overview of the major developments in this line of research including: time-frequency representations, the standard wavelet transform, directional transforms and signal dependent transforms. The chapter concludes by providing technical details of the curvelet transform, setting the stage for the next chapter where curvelet parameters are adapted.

2.1 Non-Adaptive Transforms

2.1.1 Time-Frequency Representations and Wavelets

The standard Fourier transform is well suited to study the global time-independent characteristics of signals. However, in many applications we would like to know local time dependent properties of the signal of interest. The Short Time Fourier Transform (STFT) multiplies the function to be transformed by a window, which is nonzero only over a short amount of time, to obtain the frequency content of certain locations of the signal. The Gabor transform is a special case of the STFT that uses a gaussian window function:

$$G_x(t, f) = \int e^{-\pi(\tau - t)^2} e^{-2\pi f \tau} x(\tau)\, d\tau, \tag{1}$$

where $x(\tau)$ is the signal of interest, and t and f are the time and frequency indices of the time-frequency representation. An extensive review of time-frequency representations is given in [8, 18].

One critical limitation of the Gabor transform and the STFT is the fixed scale size in the smoothing window. Wavelets are proposed as functions that provide a systematic way of dividing the data into different frequency partitions, and multiply each partition with a matched scaled window. Using wavelets, the division of frequency content depends on the frequency range. Higher frequency bands are coupled with larger smoothing windows. Biological evidence establishes connections between multi-resolution filter analysis and the visual cortex [23, 24]. Mathematically, the wavelet transform of a 1D signal $x(t)$ can be described as:

$$W_x(a,b) = \frac{1}{\sqrt{a}} \int \psi(\frac{t-b}{a})\, x(t)\, dt, \tag{2}$$

where a and b are the scale and translation parameters, respectively, and ψ is the wavelet function.

2.1.2 The Two-Dimensional Wavelet Transform

Wavelet coefficients for a given input image are computed using an iterative application of 1D low pass and high pass filters. The low passed component at each scale is used as the basis of the next iteration. Redundancy is avoided by downsampling the filtered coefficients by a factor of two. In order to ensure perfect recovery from the generated coefficients, the filters are designed to satisfy a number of mathematical properties [22, 41]. Wavelet representation allows for a form of directional selectivity as illustrated in Figure 4. Where the letters L and H in each quadrant denotes the filters used in generating the coefficients. The first letter in each expression is for the columns filter, while the second is for the rows filter. An example of applying the 2D Discrete Wavelet Transform (DWT) to a sample image is shown in Figure 5. The 2D wavelet transform is included in the JPEG 2000 image compression standard. The standard uses Daubechies 9/7 wavelets for lossy compression [4], and the 5/3 LeGall wavelet [38], which has rational coefficients, for lossless compression.

Many variants of the above algorithm have been proposed in the wavelet literature. Examples include: the dual tree complex wavelet transform [55], undecimated wavelets [58], and the use of the Laplacian pyramid [12] for scale separation.

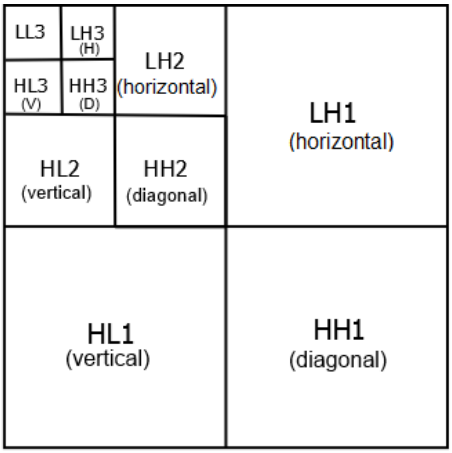


Figure 4: Dyadic wavelet decomposition and directional selectivity.

Figure 5: Wavelet decomposition of image `Cameraman`.

2.1.3 Directional Transforms

Different approaches to increase the directional selectivity of wavelet have been proposed in the literature. Gabor wavelets, for example, generalized the Gabor transform by using it in a multiscale approach with rotation parameters. The gabor wavelet function, as proposed in [39], is described in the frequency domain by:

$$\widehat{\psi}(w_x, w_{x0}, w_y, w_{y0}, \sigma, \beta, \theta) = 2\sqrt{\pi\beta\sigma}e^{-\frac{1}{2}[((w_x-w_{x0})cos\theta+(w_y-w_{y0})sin\theta)^2\sigma^2+(-(w_x-w_{x0})sin\theta+(w_y-w_{y0})cos\theta)^2\beta^2]}, \tag{3}$$

where w_x and w_y are the horizontal and vertical frequency domain coordinates, w_{x0} and w_{y0} are the centers of each Gabor tile, and σ and β are the standard deviations of the elliptical Gaussian used in forming the frequency domain divisions. Figure 6 shows an illustration of a possible tiling achieved using this scheme.

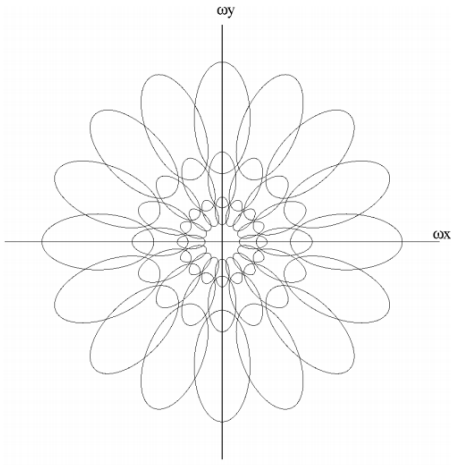


Figure 6: Gabor Wavelets and their coverage of the frequency plane.

Steerable wavelets [56, 29] capture directional information through the use of oriented partial derivative operators. The orientations are determined by angles, θ_i. Figure 7 shows a three scales steerable wavelet decomposition of a white desk.

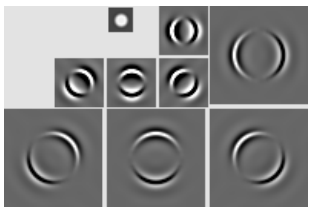


Figure 7: Steerable wavelet decomposition of a white desk with a grey surrounding.

Another class of directional transforms is motivated by recent results in approximation theory. It includes the Curvelet, Controulet [26], and Shearlet transforms [31].

The three directional transforms share an anisotropic scaling relationship, where the length of the filter's support in the frequency domain is approximately equal to the square of its width. Using dyadic decomposition the relationship between the length and the width is constructed to preserve:

$$width \approx 2^{j/2}\ ,\ length \approx 2^{j}, \tag{4}$$

where j is the scale number. Figure 8 graphically illustrates curvelets anisotropic tiling of the frequency space. The tiles shown in this figure are constructed to obey (4).

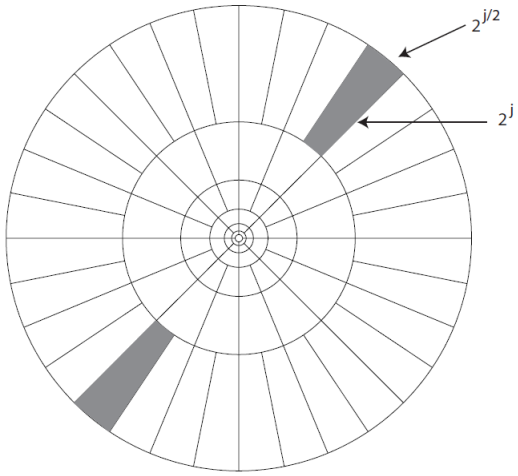


Figure 8: Curvelet's polar tiling of the frequency space.

Let the n-term partial reconstruction error be the error in approximating an image f by the largest n-term transform coefficients. Define a C^2 curve to be a curve with continuous first and second derivative. Given a smooth image with discontinuities along its C^2 edges (cartoon-like image), its n-term Fourier approximation error obeys:

$$\|f - f_n\|^2 \leq Cn^{-1/2},\ \ n \to \infty, \tag{5}$$

where C is a constant that depends on the highest magnitude transform coefficient. A wavelet representation improves the decay ratio. The approximation error using a

wavelet basis becomes:

$$\|f - f_n\|^2 \leq Cn^{-1}, \;\; n \to \infty. \tag{6}$$

In [14], the authors introduce non-adaptive curvelets and mathematically prove that the polar tiling shown in Figure 8 outperforms the wavelet decay ratio. Curvelets decay ratio was shown to obey:

$$\|f - f_n\|^2 \leq Cn^{-2} (\log n)^3, \;\; n \to \infty. \tag{7}$$

2.2 Adaptive Transforms

2.2.1 Wavelet Packets and Best Basis Selection

Coifman *et. al.* [19] introduced the notion of wavelet packets in 1992. While the multi-scale wavelet transform applies the filtering operation only to the low-passed component of each level, the wavelet packet trees of transforms include filtering operations applied to all the components generated at each level. A wavelet packet is the set of all the transforms allowed by the recursive use of DWT filtering operations. Figure 9(a) illustrates the DWT's frequency domain tiling. In Figure 9(b) an example of a possible wavelet packet tiling is shown. The division shown in Figure 9.b allows finer representation for the outer corner elements, and for scales HL1 and LH2 which were halved. Coifman *et. al.* proposed minimizing the entropy of wavelet coefficients as a cost function guiding the selection of the best wavelet basis. The adaptivity provided by wavelet packets is limited, due to the non-directionality of the transform and the fixed scale locations, which we define as the boundary points that determine where each frequency division starts or ends.

2.2.2 Pre-and post-transform adaptive operations

Several authors introduced adaptivity by operations applied prior and after the division of signal content into unique subbands. For example, the Easy Path Wavelet Transform (EPWT) [50] divides the input image into 1D paths chosen to maximize the

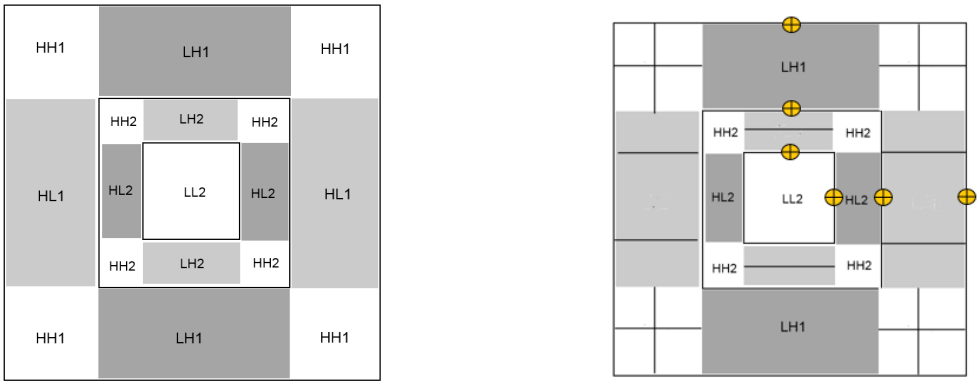


Figure 9: Wavelet and wavelet packets coverage of the frequency domain: (a) Wavelet transform tiling and (b) a possible wavelet packet tiling. Scale locations are denoted by the yellow markers,⊕.

correlations between its data points. Next, the transform applies a standard wavelet transform to the acquired paths. In [16], the arrangement of inputs to the wavelet transform is designed to minimize a coding cost function. Similar pre-transform adaptive ideas are proposed in [43, 36]. Bandlets [42] present an example of post-wavelet adaptation, where the geometric flow of wavelet coefficients is parameterized.

It is worth noting, that the algorithms belonging to the categories presented in this section can possibly benefit from the adaptive scheme presented in this dissertation. For example, EPWT wavelet's machinery handles vectors with highly correlated entries and a better representation of such vectors can be obtained using customized frequency domain tilings.

2.3 The Curvelet Transform

2.3.1 Flow of the Algorithm

The first step in computing the Curvelet transform of in image is to apply the 2D fast Fourier transform (FFT). The Fourier plane is constructed so that the center frequency (*i.e.* D.C. value) is located at the center of the plane. Next, the FFT plane is divided into tiles as shown in Figure 10. The inner most level (Scale 0) is not

directional. Periodic extension is used in the outer scale to reduce boundary artifacts. In this figure, the plane is divided into five scales with Scale 0 covering the lowest frequency components.

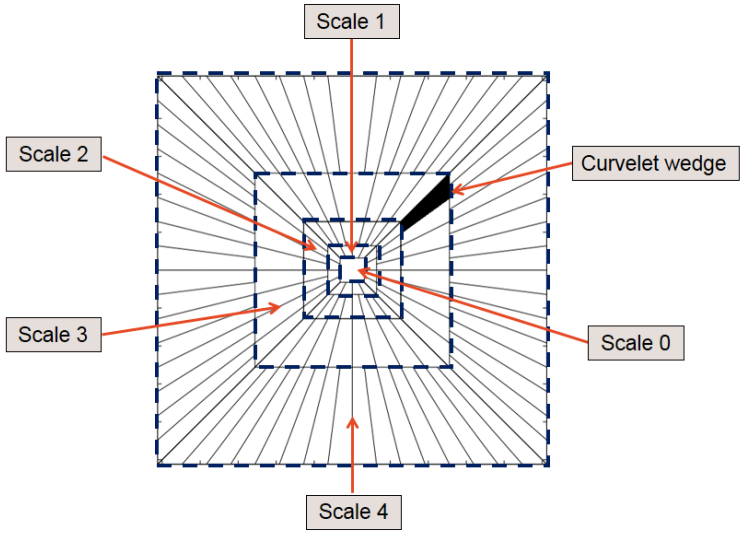


Figure 10: Default curvelet tiling. A five scales decomposition is shown. Each scale is divided into $16 \times 2^{\lceil (j-1)/2 \rceil}$ wedges, where j is the scale number.

To reduce boundary artifacts each curvelet wedge is smoothed in the four directions. The smoothing is performed in an overlapping manner, where each smoothing function covers two neighboring wedges. This overlapping scheme is used to reduce blocking artifacts. The smoothed region belonging to two neighboring wedges, includes all Fourier transform plane coordinates smoothed by the smoothing functions. Each Fourier transform coefficient value p in the smoothing region between two wedges is multiplied by smoothing functions U_1 and U_2 (Figure 11). pU_1 is used in computing the coefficients representing the first wedge. Similarly, pU_2 is used in computing the coefficients representing the second wedge.

Perfect recovery from the coefficients is made possible by ensuring that the smoothing functions are normalized, so that the following property holds [13]:

$$U_1^2 + U_2^2 = 1. \tag{8}$$

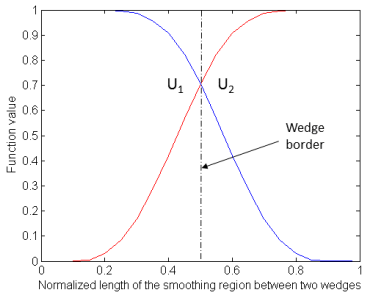


Figure 11: Two complementary smoothing functions U_1 and U_2 shown in red and blue respectively.

Curvelet coefficients are computed by taking the inverse FFT of each smoothed wedge. Taking the inverse FFT on these non-rectangular wedges can be performed by a "wrapping" operation. In this step, every wedge is localized inside a parallelogram. Figure 12 shows a smoothed curvelet tile along with its parallelogram-shaped support region.

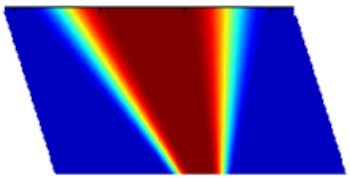


Figure 12: A smoothed curvelet tile shown with its parallelogram-shaped support region. Function values decrease from 1 to 0 as color changes from red to blue.

Using the periodicity of the Fourier transform, the 2D plane is tiled with copies of this parallelogram in a manner similar to what is shown in Figure 13. Note that the same plane tiling can equivalently be generated using copies of the rectangular region surrounding the origin. This allows us to compute the inverse FFT of the parallelogram-shaped region by computing the inverse FFT of the rectangle surrounding the origin with the parallelogram's length and height.

Another possible method for taking the inverse FFT involves using a shear and an interpolation procedure and is detailed in [13]. The inverse curvelet transform works by "reversing" forward transform operations. The flow of the forward and inverse

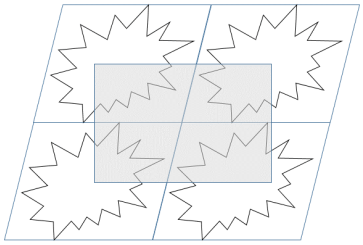


Figure 13: The inverse FFT of the parallelogram-shaped regions is computed by taking the inverse FFT of the shaded rectangular region surrounding the origin.

curvelet algorithms is summarized in Figure 14.

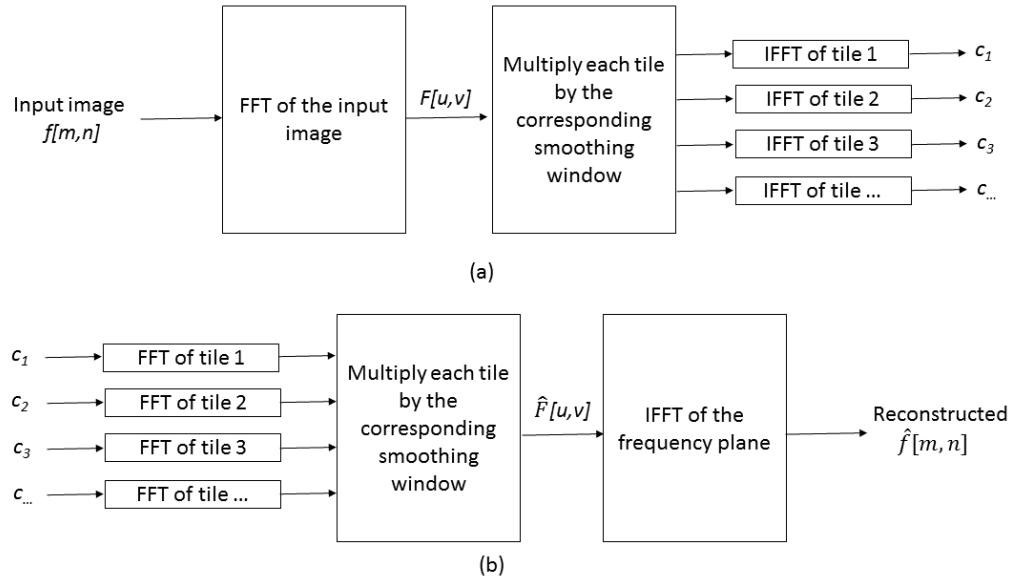


Figure 14: Summary of (a) Forward curvelet transform (b) Inverse curvelet transform.

Contourlets [26] and Shearlets [31] provide alternate implementations of the anisotropic tiling relationship shown in (4). Contourlets implements a curvelet-like tiling of the frequency domain using filtering and downsampling operations in the spatial domain. The Shearlet transform is based on the pseudo-polar DFT, which allows a convenient method for taking the inverse Fourier transform of directional tiles (*i.e.* eliminating the wrapping procedure).

2.3.2 Default Parameter Values

Scales are distributed in a dyadic manner where each scale is larger in size than the previous one by a factor of two. The outer most level is chosen to be one third of the size of the image. The number of scales J is dependent on image dimensions N_1 and N_2, and is given by

$$J = \lceil \log_2(\min(N_1, N_2) - 3) \rceil, \tag{9}$$

where $\lceil * \rceil$ is the ceiling function. The number of divisions per scale is equal to $16 \times 2^{\lceil (j-1)/2 \rceil}$, where j is the scale index. In the next chapter, different strategies will be used to adapt default curvelet parameter values with the goal of finding the set of parameters that maximizes a given cost function.

CHAPTER III

ADAPTIVE CURVELETS

Adaptive curvelets alter the number, size, and locations of curvelet tiles according to a given cost function. The frequency-domain tiling that generates the best improvement in the cost function value is considered optimal. In denoising-based adaptive curvelets the cost function is denoising performance as measured by PSNR or a similar performing measure. Additive White Gaussian Noise (AWGN) with zero mean and standard deviation σ is added to the input image. Next, a value for the number decomposition scales is chosen by the proposed scale selection algorithm. This will be followed by search algorithms finding the optimal scale locations and number of angular divisions for each scale/quadrant pair. The algorithms converge returning the optimal curvelet tiling for the given image.

The optimization problem characterizing denoising-based adaptive curvelets can be described mathematically by

$$\arg\max_{S,A} PSNR(I, \hat{I}_{D(S,A)}), \tag{10}$$

where I is the input image, $\hat{I}_{D(S,A)}$ is the denoised image. This image is obtained using adaptive curvelets described by parameters S and A, S is a $2 \times J$ matrix describing adaptive curvelet scale locations, A is a $2 \times J$ matrix that determines the number of angular decompositions used in each curvelet scale/quadrant pair. J is the number of scales used in curvelet decomposition. S and A matrices have two rows referencing the two curvelet quadrants over which computation of curvelet coefficients is performed. $PSNR(I, \hat{I})$ is function that computes the peak signal to noise ratio between images I and $\hat{I}$. The denoising algorithm used in this work is described in the next section. This is followed by sections describing the algorithms used to search

for a solution of (10).

3.1 The Cost Function: Denoising Performance

A number of methods to guide the choice for optimal transform parameters were recently proposed in the literature. For example, Grouplets [43], Direction-adaptive wavelets [16], and Tetrolets [36] search for parameters that minimize the l_1-norm of transform coefficients. This approach is not applicable to our problem since adapting curvelet tiles will change the total number of coefficients. The l_1-norm is not defined between vectors with different lengths. The proposed cost function relies instead on denoising performance.

Using denoising performance as a cost function provide a number of advantages. First, it is a function that has direct connection to the quality of the reconstructed image. This is a desired feature that can, as will be shown in this dissertation, permits the use of denoising-based tilings in a number of other applications. Curvelet-based denoising is also an efficient function that runs in about 1-2 seconds in modern computers. This low computational cost reduces the overall running time of the search algorithm.

In order to apply the cost function, the input image is contaminated with additive white gaussian noise (AWGN) with zero mean and standard deviation σ. Guidelines for choosing the optimal value of σ will be provided in Section 3.7. Denoising performance is measured by PSNR or the logarithm of the mean squared error (log (MSE)). In general, curvelet tilings found by the search algorithm did not change much by varying the amount of noise used in searching for the optimal curvelet tiling. However, the value of σ must not overpower the underlying image or be negligible when compared to image variations.

Similar to wavelets, Gaussian noise can be eliminated in the curvelet domain using thresholding methods. The thresholding algorithm used in this work [59] applies a

simple monte-carlo simulation to estimate a wedge dependant noise standard deviation from the estimate of noise standard deviation σ corrupting the original image. The local noise level for wedge$_{j,l}$ $\sigma_{j,l}$ for scale decomposition S and angular decomposition A is estimated by applying the curvelet transform with the same scale and angular decomposition to a white noise image with σ noise level. The wedge dependant noise standard deviation is computed using the generated coefficients. This local noise estimation can be repeated for a few more iterations to reach a reliable estimate. Let ĉ be the noisy curvelet coefficients. Restored image coefficients c are given by hard-thresholding according to the following equation:

$$c = \widehat{c}, \qquad \text{if } |\widehat{c}| \geq k\sigma_{j,l} \tag{11}$$

$$c = 0, \qquad \text{if } |\widehat{c}| < k\sigma_{j,l}, \tag{12}$$

where $k = 4$ for the outer scale, and is equal to three otherwise. k values were determined experimentally. Varying these values allows for a balance between denoising blurring artifacts and accurate noise removal.

3.2 Scale Selection Algorithm: Number of decomposition scales

A connected region of high magnitude FFT values is observed in the center of the frequency plane of most natural images. An example of such activity is shown in Figure 15. The figure shows a plot of FFT log-magnitudes for two images. FFT values decrease as color changes from red to blue.

To achieve a better representation of the signal of interest, it is preferable to avoid dividing such connected regions into tiles while applying the curvelet transform. We developed a heuristic algorithm to determine the number of scales guided by this observation. The algorithm works as follows:

Step 1 define a quantity called mid-range mR, as follows:

$$mR \triangleq \sqrt{\max |FFT\ value| \times \min |FFT\ value|}$$

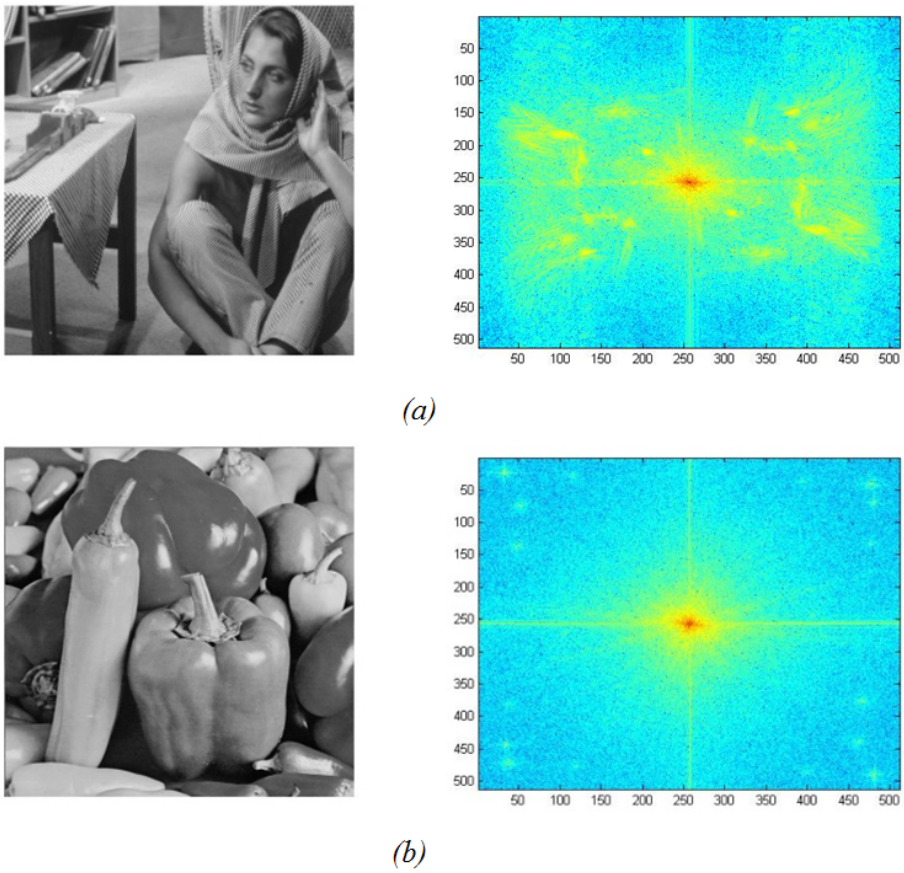


Figure 15: `Barbara` and `Peppers` images and their corresponding FFT log magnitude plots. FFT values decrease as color changes from red to blue.

Step 2 let $D = D_{initial} = 5$. This is the smallest possible value for the size of the inner most curvelet level.

Step 3 set a square surrounding the origin of the frequency domain with a diagonal length equal to D pixels

Step 4 if any FFT magnitude value within the square is smaller than the mid-range value, exit the algorithm and return $D = D - 2$. Otherwise, go to step 5

Step 5 set $D = D + 2$ and return to step 3

The optimal number of decomposition scales is found using the value D computed in step 4 above. Using dyadic scaling the optimal number of scales J is computed as

follows:

$$J = \lceil (\log_2(\max(N_1, N_2) - 1) - (\log_2 D - 1) \rceil \tag{13}$$

$$= \lceil (\log_2(\max(N_1, N_2)) - (\log_2 D) \rceil \tag{14}$$

$$= \left\lceil \log_2(\frac{\max(N_1, N_2)}{D}) \right\rceil, \tag{15}$$

where N_1 and N_2 are the image vertical and horizontal dimensions respectively. Equation (13) computes the difference between the number of scales needed to reach the origin of the frequency domain from the edge of the image, and the number of scales needed to reach the origin from the edge of the coarsest level square. The operator $\lceil * \rceil$ is the ceiling function.

Seismic and finger print data are two examples of images that do not exhibit the centered impulse behavior. A small value of D can detect such cases, where equation (9) is used instead for selecting the number of scales.

This algorithm was tested on 500 test images acquired by crawling the website DPChallenge.com. Four hundred and sixty five images were found to have a centered impulse region. Those images were corrupted by additive white Gaussian noise (AWGN) with σ=25. A sample of these images is shown in Figure 16.

The scale selection algorithm generates denoising improvements in 83.0% of cases. In images where adaptive curvelets outperform default curvelets, the average improvement in PSNR is 0.20dB . Average PSNR improvement over all images is found to equal 0.095dB. The number of scales chosen by the scale selection algorithm is going to be used as the basis for further improvements achieved by the remaining parts of the optimization algorithm.

It is noteworthy that the size of the centered impulse response is observed to have a relationship with image quality. For many distortion types, distorted images typically have a smaller impulse region size. Figure 17 shows a sample image from the 2008 Tampere Image Database (TID) [51] along with two distorted versions of

Figure 16: Sample of the images used to test the scale selection algorithm shown with the corresponding D values and image sizes.

the image. Figure 17(b) show the sample image corrupted with spatially correlated noise. In Figure 17(c) the sample image is corrupted with quantization noise. In the original image, the size of the centered impulse region D is equal to 25. As shown in Figure 17, the addition of noise reduces the value of D to 23 for the spatially correlated noise image. The value of D is reduced to 19 in the quantization noise image.

The decrease in the size of the centered impulse region is noticed with other types of noise used in the TID database. The value of D is reduced for images contaminated with additive Gaussian noise, masked noise, impulse noise, JPEG transmission errors, JPEG 2000 transmission errors, and the non-eccentric pattern noise. These experimental results show that the value of D and possibly other characteristics of the centered impulse region are promising tools for image quality assessment.

An algorithm to compute the number of scale decompositions in adaptive curvelets was presented in this section. Next, an algorithm that optimizes the number of angular divisions per scale/quadrant pair is presented.

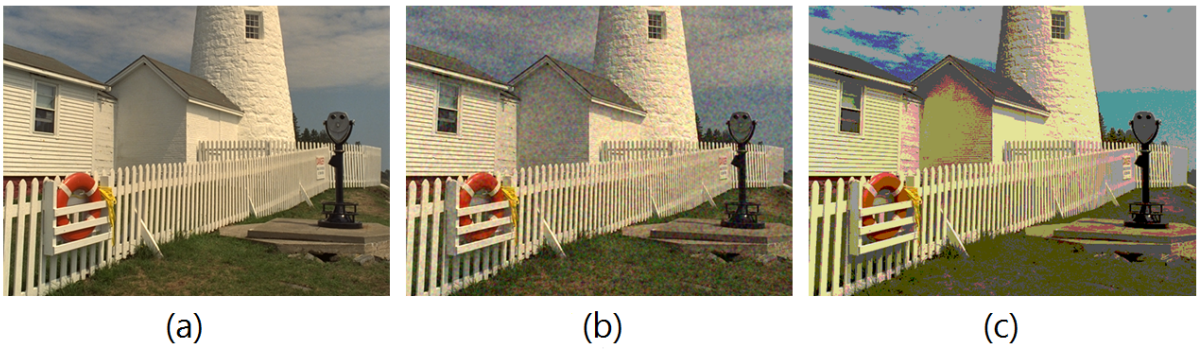


Figure 17: The effect of noise on the size of the centered impulse region D. (a) Sample image $D = 25$. (b) Image "a" corrupted with spatially correlated noise $PSNR = 27.04$, $D = 23$. (c) Image "a" corrupted with quantization noise $PSNR = 21.10$, $D = 19$.

3.3 Optimal Number of Angular Divisions per Scale/Quadrant

Optimizing the number of angular divisions is performed in a brute force fashion in each curvelet scale/quadrant pair (Figure 18). The tested parameters are chosen from the following sequence {4, 8, 12, 16, 20, 24}. The number of divisions that achieve the maximum cost function value is considered optimal. The divisions are uniformly distributed in each scale/quadrant pair. Recalling FFTs symmetry for real data the number of parameters to optimize is $2\times(J-1)$ for real data, and is equal to $4\times(J-1)$ for the case of complex input data.

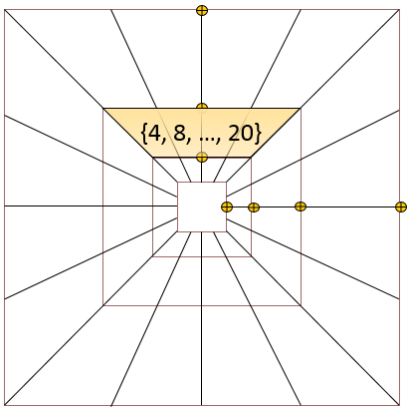


Figure 18: Optimal angular decomposition is found through a brute-force search strategy for each scale/quadrant pair. The figure shows the Second scale/first quadrant curvelet segment.

3.4 Optimal Scale Locations

We propose using derivative-free optimization methods to search for adaptive curvelets scale-locations. Such methods start at an initial point x_0, evaluate the cost function at a selected mesh of points in the neighborhood of x_0, and define the point in the mesh with the minimum function value as the new x_0. Next, the algorithm iterates until a specified convergence criteria is met. The Nelder-Mead simplex search method [48, 20] is one of the popular methods for generating such a mesh. It has been used extensively in a variety of application areas. It is used in this work to find the optimal scale locations.

Let $\{V_1, V_2, ..., V_J\}$ be the coordinates of vertical scale locations, and $\{H_1, H_2, ..., H_J\}$ be the horizontal scale locations (Figure 19). Curvelet scales are not constrained

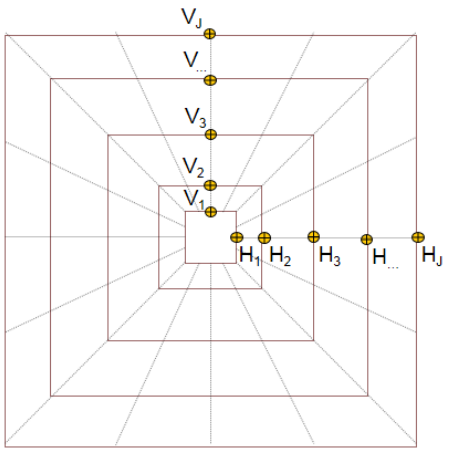


Figure 19: Vertical and horizontal scale locations.

to be of the same length and width. This makes the number of optimizing parameters equal to $2\times J$ for real data. To ensure meaningful choices for scale locations, the simplex method optimizes a constraint checking function that returns a PSNR value equal to zero whenever any of the constraints is violated. Otherwise, it returns the denoising PSNR for the requested parameters (Figure 20).

The enforced constraints are:

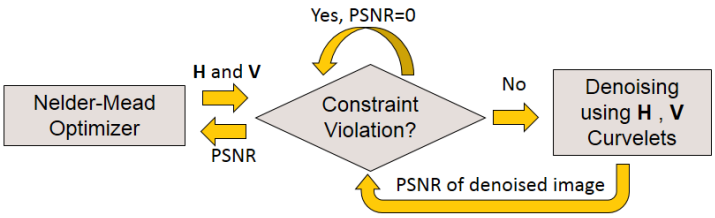


Figure 20: Scale locations optimizer and the constraint checking function.

(i) The inner scale locations are above and to the right of the periodically extended image center: $V_1 > \lfloor V_J\ /2 \rfloor$, and $H_1 > \lfloor H_J/2 \rfloor$.

(ii) Scales are organized in an ascending order: $V_j > V_{j-1}$, and $H_j > H_{j-1}$.

(iii) Periodic extension is used in the outer level J to enhance curvelet's performance by connecting border elements. Figure 21 graphically illustrates the extension region. By definition, the outer border of this level needs to be outside the image boundaries: $V_J > N_1$, and $H_J > N_2$.

(iv) Level *J-1* is constrained to lie inside the original image boundaries. This constraint is used to avoid dividing the periodic extension level by moving this level outside the image boundaries: $V_{J-1} \leq N_1$, and $H_{J-1} \leq N_2$.

The algorithm iterates until convergence or until a maximum number of iterations is reached. More details about the Nelder-Mead algorithm are provided in Section 3.8.

3.5 Size of Overlap Regions between Scales

Curvelet wedges are smoothed in the four boundary directions. The smoothing operation limits discontinuities across wedge borders. The smoothing is either within the current scale, or across different scales. Adapting curvelet tilings will change scale locations. The inter-scale smoothing window sizes need to change accordingly. Smoothing windows length determine the amount of overlap between neighboring scales.

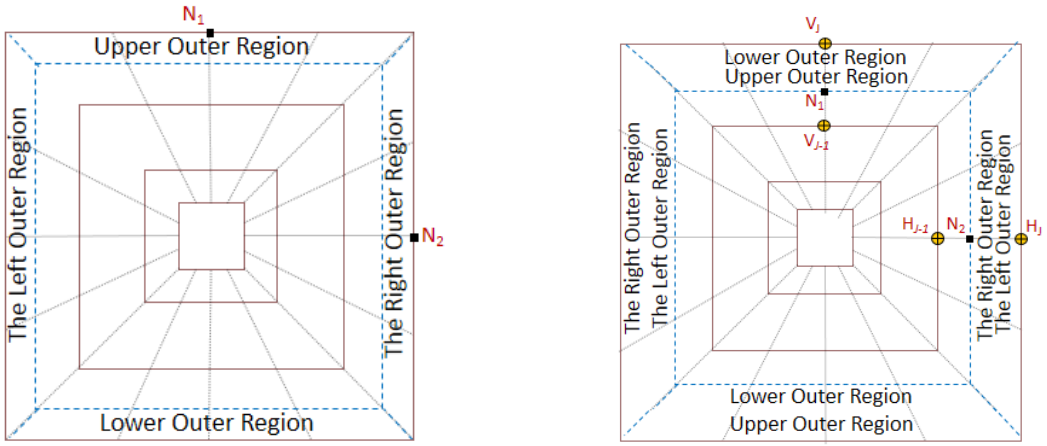


Figure 21: Curvelet's outer periodic extension region: (a) Fourier Transform (FT) plane before adding the extension region to an image of size N_1, N_2 (b) FT plane after the extension.

Based on our experimental results, the following equation is chosen to determine the length of horizontal or vertical overlap region between scales:

$$Overlap(j) = \begin{cases} \min(0.8\max(\Delta_{j+}, \Delta_{j-}), \min(\Delta_{j+}, \Delta_{j0})), & \text{for } 1 < j < J \quad (16) \\ \min(0.8\max(\Delta_{j+}, \Delta_{j0}), \min(\Delta_{j+}, \Delta_{j0})), & \text{for } j = 1, \quad (17) \end{cases}$$

where $j \in \{1, 2, 3, ..., J-1\}$ is the scale number, Δ_{j+} is the distance between scale *j+1* and scale j, Δ_{j-} is the distance between scale j and scale *j-1* , and Δ_{j0} is the distance between scale j and the FFT plane center. The quantities Δ_{j+}, Δ_{j-}, and Δ_{j0} are shown on a curvelet grid in Figure 22.

The term $\min(\Delta_{j+}, \Delta_{j0})$, in (16) and (17), sets an upper bound on the size of smoothing windows. Figure 23 graphically illustrates the behavior of (16) for the case where no upper bounds are violated.

Figure 24(a) shows an example where the bound Δ_{j0} is invoked due to a large outer scale size. In this figure, the outer distance Δ_{j+} is much larger than the inner distance Δ_{j-}. If Δ_{j+} was used as a length of the smoothing window, the smoothing operation would extend to Fourier plane coordinates to the left of the image center.

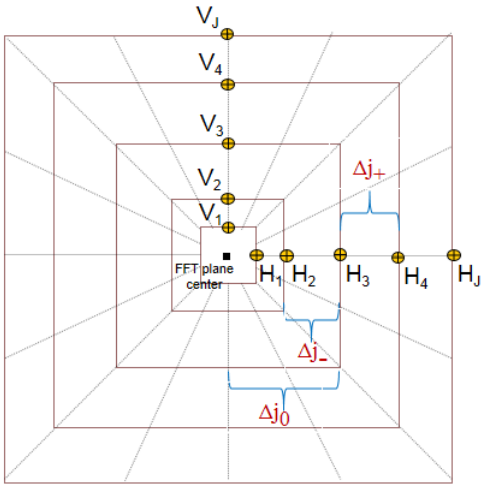


Figure 22: Quantities used in determining the length of the overlap region in scale location H_3.

This situation is avoided by using the upper bound Δ_{j0} as a length for the smoothing widow.

Similarly, Figure 24(b) shows an example where Δ_{j+} is needed for the case of a large inner scale size. In this figure, the inner distance Δ_{j-} is much larger than the outer distance Δ_{j+}. If Δ_{j-} was used as a length for the smoothing window, the smoothing operation would extend to Fourier plane coordinates outside $j = 3$ right's border. According to the current implementation of curvelets, the coefficients belonging to this outer scale are computed prior to the computation of smoothing window size for level $j = 2$. Multiplying these Fourier transform coordinates by the smoothing window is no longer possible. This situation is avoided by using the upper bound Δ_{j+} as a length for the smoothing window.

3.6 Global Optimization

In the previous section, three separate algorithms for finding the optimal number of scales, scale locations, and angular divisions were presented. In this section, we turn our attention to the development of a global optimization algorithm combining the previous adaptations. The algorithm uses a multi-resolution search strategy. This

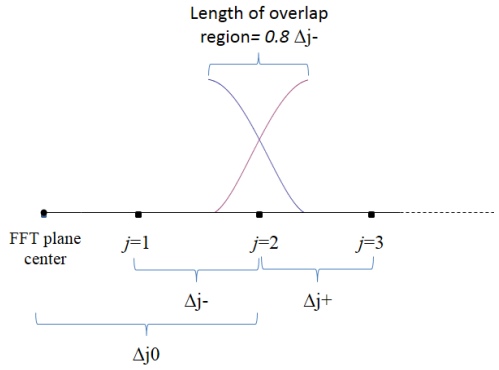


Figure 23: Length of the overlap region in scale j=2 as determined by equation (6).

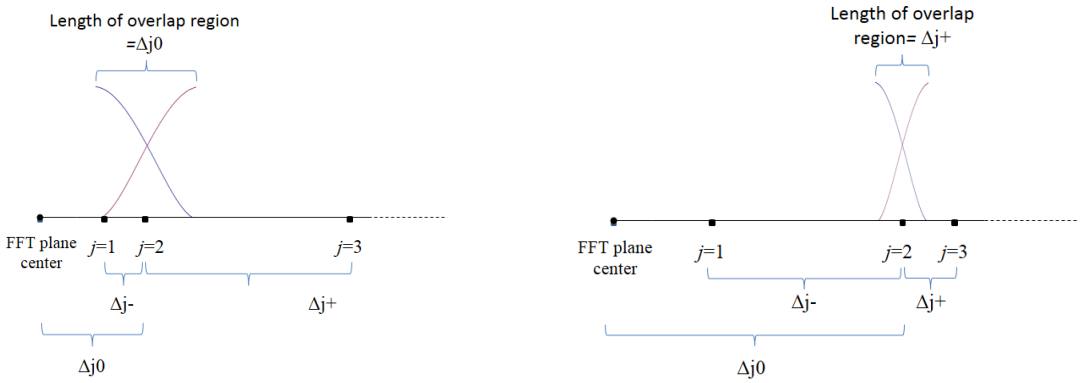


Figure 24: Bounds enforced by equation 16 on the length of the overlap region: (a) Using the upper bound because $\Delta_{j+} > \Delta_{j0}$ (b) Using the upper bound because $\Delta_{j-} > \Delta_{j+}$.

helps in avoiding convergence to local minima, and reduces the computational cost of the algorithm. Optimization is done in hierarchical manner consisting of n iterations. In each iteration, optimal scale and angular locations are found for a downsampled smoothed version of the input image. The flow of this algorithm is illustrated in Figure 25.

The algorithm starts with finding the optimal number of scales J using the scale selection algorithm (Section 3.A). Next, the original image is downsampled $n-1$ times by a factor of 1/2. The image is also smoothed by a Gaussian filter to prevent aliasing artifacts. The optimal angular decomposition is found using dyadic scale locations and the optimal number of scales computed. The angular decompositions found and

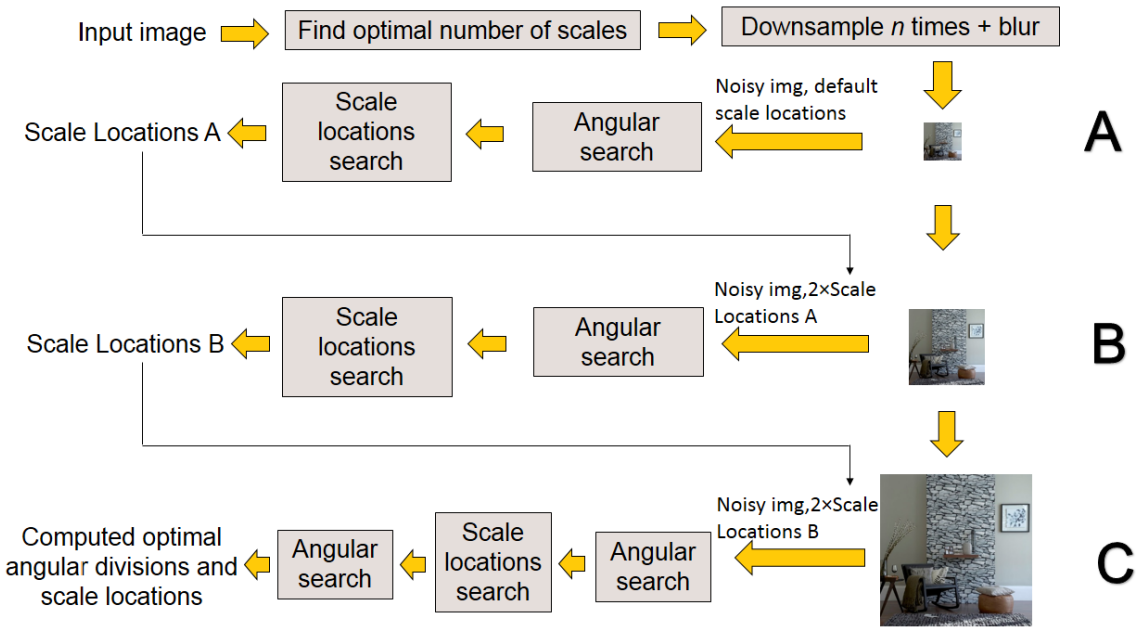


Figure 25: Flow diagram of the global optimization algorithm.

the dyadic scale locations are used as initial points for the scale locations search algorithm. Next, the value of n is decremented by one. and a new loop starts. Initial values for scale locations will be the scale locations computed in the previous iteration after being rescaled to the new image coordinates. After n iterations, once the scale locations search algorithm converges to a minimum, the angular decomposition algorithm is run to search for the optimal angular decomposition at the calculated scale locations. Finally, the algorithm terminates returning the determined optimal scale and angular decompositions.

3.7 Choosing a standard deviation value σ for the added noise

The search for optimal curvelet tilings, given the proposed denoising-based objective function, is dependant on the proper choice for the added noise standard deviation σ. Since the added Gaussian noise is zero mean, the effect of the noise on image content can be captured by the standard deviation of image intensities. A low value for the ratio between σ and the image standard deviation will cause the optimization

algorithm to converge to default scale locations. On the other hand, a high value for σ can overpower image content (*i.e.* details of the image will disappear) and lead the optimization algorithm to arbitrary solutions. These solutions will depend primarily on the noise pattern added to the signal.

In practice, a wide range of σ values are tolerable by the optimization algorithm. In denoising and sparse recovery of seismic datasets, changing the value of σ within the interval $1/2 \times \sigma_{img} \leq \sigma_{training} \leq \sigma_{img}$ had marginal effect on the results found.

A sample face image extracted from the AT&T database is shown in Figure 26. The figure also shown a plot of the relative imrpvoement over default curvelets S defined as

$$S_{\sigma_i} = \frac{MSE_{default} - MSE^{\sigma_i}_{adaptive}}{MSE_{noise}}, \tag{18}$$

where $MSE_{default}$ is the mean squared error between the image denoised using default curvelets and the original image. $MSE^{\sigma_i}_{adaptive}$ is the mean squared error between the image denoised using adaptive curvelets and the original image. $MSE^{\sigma_i}_{adaptive}$ is generated using σ_i as the standard deviation of the added Gaussian noise. MSE_{noise} is the mean squared error between the noise image and the original image. S values are plotted across a wide range of σ_{train} values.

At high values of noise improvements caused by adaptive curvelets reach zero. A similar behaviour is observed at very low σ values. The peak S value occurs at a σ value of 3.2dB. Similar curves are observed with the remaining datasets used in this dissertation.

3.8 Nelder-Mead's Convergence and Computational Complexity

Convergence of the proposed method relies on the convergence of the Nelder-Mead optimization algorithm. In this section, details on this algorithm are given along with with the convergence criteria. The algorithm starts by forming a simplex of size $n+1$,

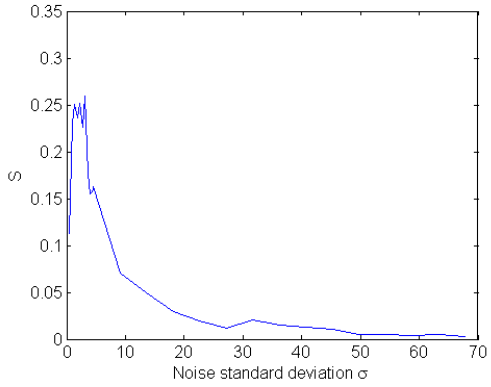


Figure 26: A sample image and its corresponding S-value plot.

where n is the dimension of the vector to be optimized. Vertices of this simplex will be placed at the initial point x_0 and at n other points formed by increasing the value of each element of x_0 by 5%. The cost function is evaluated at each of the vertices and the worst vertex is rejected. At each iteration the Nelder-Mead algorithm performs the following sequence of operations to determine a new vertex:

1. Compute function values at each vertex. Sort function values so that $f(1) \leq f(2) \leq ... \leq f(n+1)$. Assuming a minimization problem $f(n+1)$ becomes the worst vertex in the simplex. Compute the center of mass for the first n points $\bar{x} = \sum_i^n x_i/n$.

2. Compute the function value at the reflection point R. This point is the reflection of $f(n+1)$ around $\bar{x}$. if $f(1) \leq f(R) < f(n)$ replace vertex $n+1$ with R. if $f(R) < f(1)$ perform an expansion step and compute the function value at $f(E)$. Figure 27 graphically illustrates the reflection and expansion steps for a 2D problem.. If $f(E) < f(R)$ replace vertex $n+1$ with E. Otherwise, use vertex R. Finally, if $f(R) \geq f(n)$ move to the next step.

3. Perform a contraction with the minimum of $f(R)$ and $f(n+1)$. if R was found to be the minimum the contraction is outside the original simplex. Otherwise, the contraction is within the simplex. Contraction operations are illustrated in

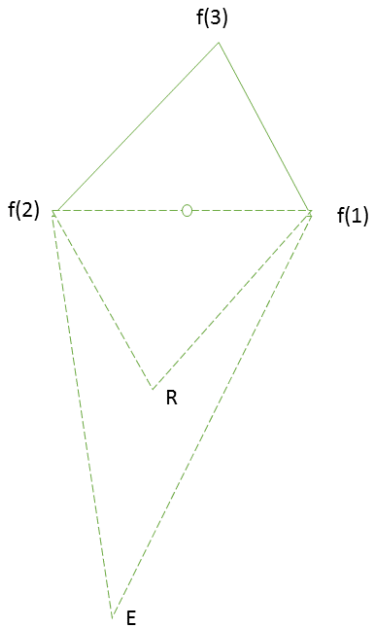


Figure 27: Graphical illustration of the reflection and expansion steps in the Nelder-Mead algorithm.

Figure 28. If $f(C_{in}) < f(n+1)$ accept $f(C_{in})$. Otherwise, move to the next step. Similarly, if $f(C_{out}) < f(R)$ accept $f(C_{out})$. Otherwise, move to the next step.

4. Shrink the simplex: $f(1)$ remains fixed. Other vertices move so that their distance to $f(1)$ is shortened by $1/2$. The operation is illustrated in Figure 29.

Nelder-Mead's implementation was setup so that convergence occurs when the maximum difference between scale locations falls below one pixel or when a maximum number of iterations is reached. The maximum number of iterations was set to equal 500.

The developed global search algorithm depends on repeated application of curvelet denoising operations. Assuming an image of size $n \times n$, the forward and inverse curvelet transforms have a complexity in the order of $O(n^2 \log n)$. The thresholding operation complexity is in the order of $O(n^2)$. This makes the total complexity of the search procedure equivalent to $O(n^2 \log n)$.

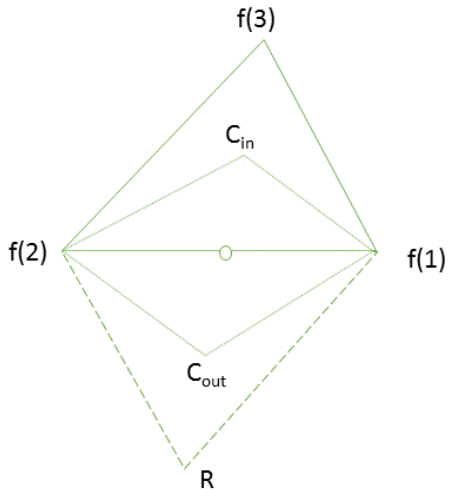


Figure 28: Graphical illustration of the contraction step in the Nelder-Mead algorithm.

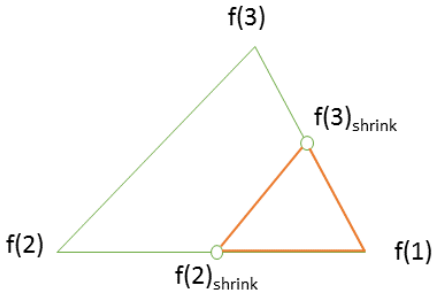


Figure 29: Nelder-Mead's shrink operation.

Actual computational times for a sample of images with sizes 512×512 and 256×256 are shown in Table 1. The table also shows the number of cost function evaluations. The number of cost function evaluations includes cost function calls by the Nelder-Mead algorithm and the evaluations made using the angular search algorithm. The experiments were run using an Intel i7-4500 processor operating at 1.80GHz with a maximum frequency of 3.00GHz. The machine has 8.00GB of installed memory. The search algorithm was built using MATLAB's implementation of Nelder-Mead.

Table 1: Time and total number of cost function evaluations used in computing denoising-based adaptive curvelets for images `Barbara`, `Bird`, and `Cameraman`.

	Image size	Elapsed time (min.)	Number of function evaluations
`Barbara`	512×512	15.0	1039
`Bird`	512×512	12.7	1042
`Cameraman`	512×512	12.1	949
`Barbara`	256×256	3.8	917
`Bird`	256×256	3.1	925
`Cameraman`	256×256	3.3	860

3.9 Coefficient Decay and Partial Reconstruction

In this section, we present a sparsity analysis of adaptive curvelet coefficients. Adaptive curvelet tiles were learned using AWGN with $\sigma = 25$. Magnitude plots of curvelet and adaptive curvelet coefficients for three different images of size 512×512 are shown in Figure 30(b). Adaptive curvelets are shown to reduce the magnitudes of curvelets coefficients at the expense of generating a higher number of them. Comparisons between default and adaptive curvelets partial reconstruction error is presented in Figure 30(c). The figure plots the PSNR of images reconstructed using the highest magnitude curvelet coefficients. The number of curvelet coefficients used is shown as a percentage of the number of image pixels. Adaptive curvelet coefficients show improved performance in mid and low quality compression regions.

Default curvelets are reported in the literature to generate improved partial reconstruction performance over wavelets [13, 32]. The results shown in this section indicate that adaptive curvelets are promising candidates for the development of competitive image compression systems.

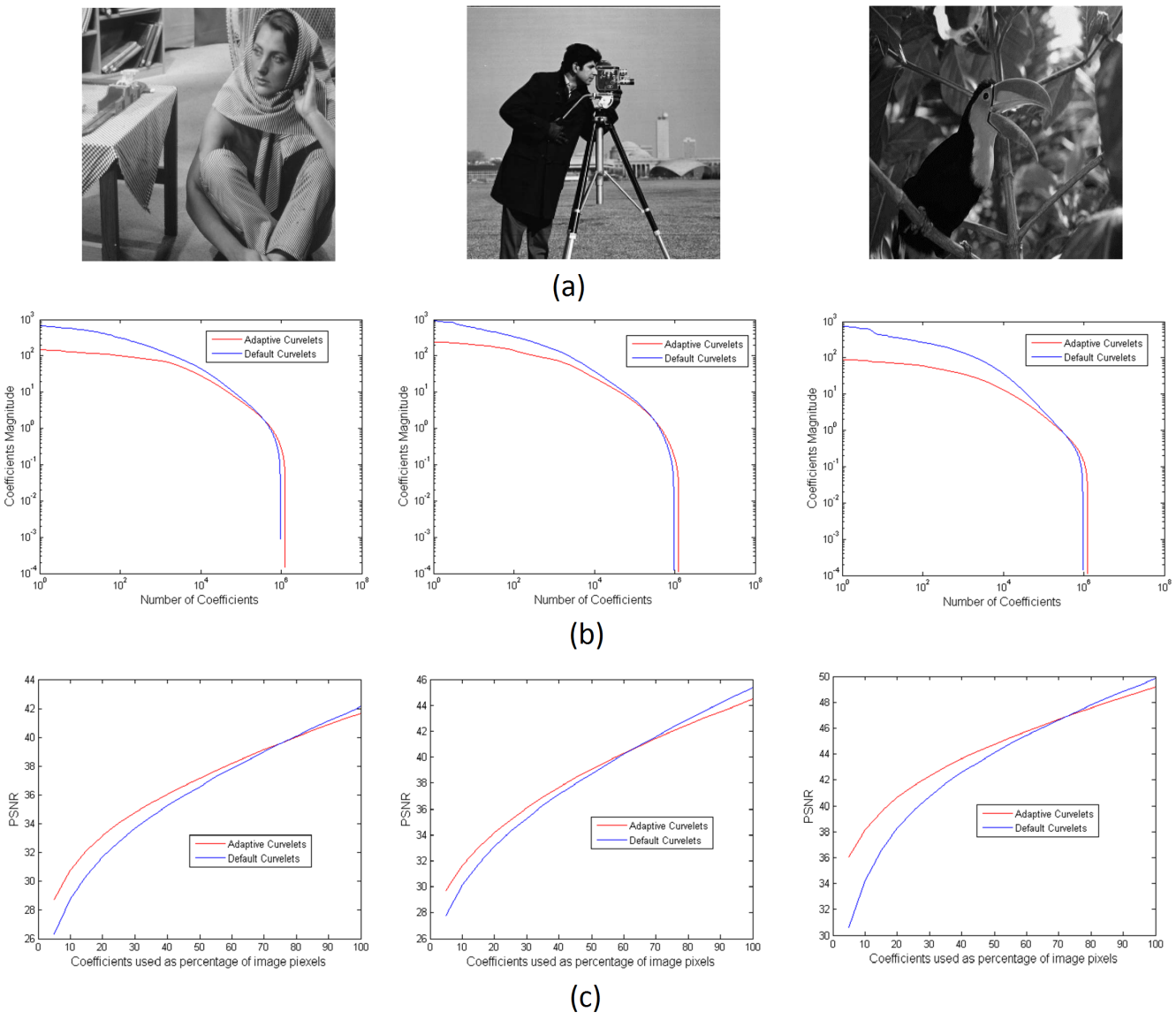


Figure 30: (a) Images `Barbara`, `Cameraman`, and `Bird`. (b) Default and adaptive curvelet decay curves shown on a log-log scale. (c) Partial reconstruction error from the highest curvelet coefficients. The number of coefficients used is shown as a percentage of the number of image pixels 512×512.

3.10 Training Curvelets

In certain applications, access to the original data is not available. For example, the dataset could be subsampled or corrupted by occlusions or corrupted by noise. In such cases, curvelet tiles can be learned from a group of images representing the specialized class to which the dataset of interest belongs. Examples of these classes include: medical images, Synthetic aperture radar (SAR) data, and seismic data sets. Figure 31 illustrates an example of two classes of images and their unique fingerprints in the 2D Fourier transform plane. Fourier activity in face images is distributed around the center of the Fourier plane. While in seismic data, the activity is clustered into two groups separated by a narrow horizontal gap. Training curvelets to learn tiles that are representative of the class of interest will result in improved data representations.

In what follows, an algorithm to learn a curvelet representation that better represents a class of images is presented. First, the scale selection algorithm is used to find the number of scale decompositions in each image in the training dataset. The average over all images is rounded to the nearest integer and is used as the number of scale decompositions. Next, a search for the optimal tiling is run for each image in the training dataset using the averaged number of scale decompositions. The average scale locations and angular decompositions, rounded to the nearest integers, is considered the tiling representative of the class of interest. In our experiments, the average angular decomposition was obtained by rounding to the nearest multiple of four since the implementation of adaptive curvelets developed admits only multiples of four as possible values for the number of angular decomposition elements. In the next two sections, we present experiments on denoising seismic and face images using curvelet tiles learned by this algorithm.

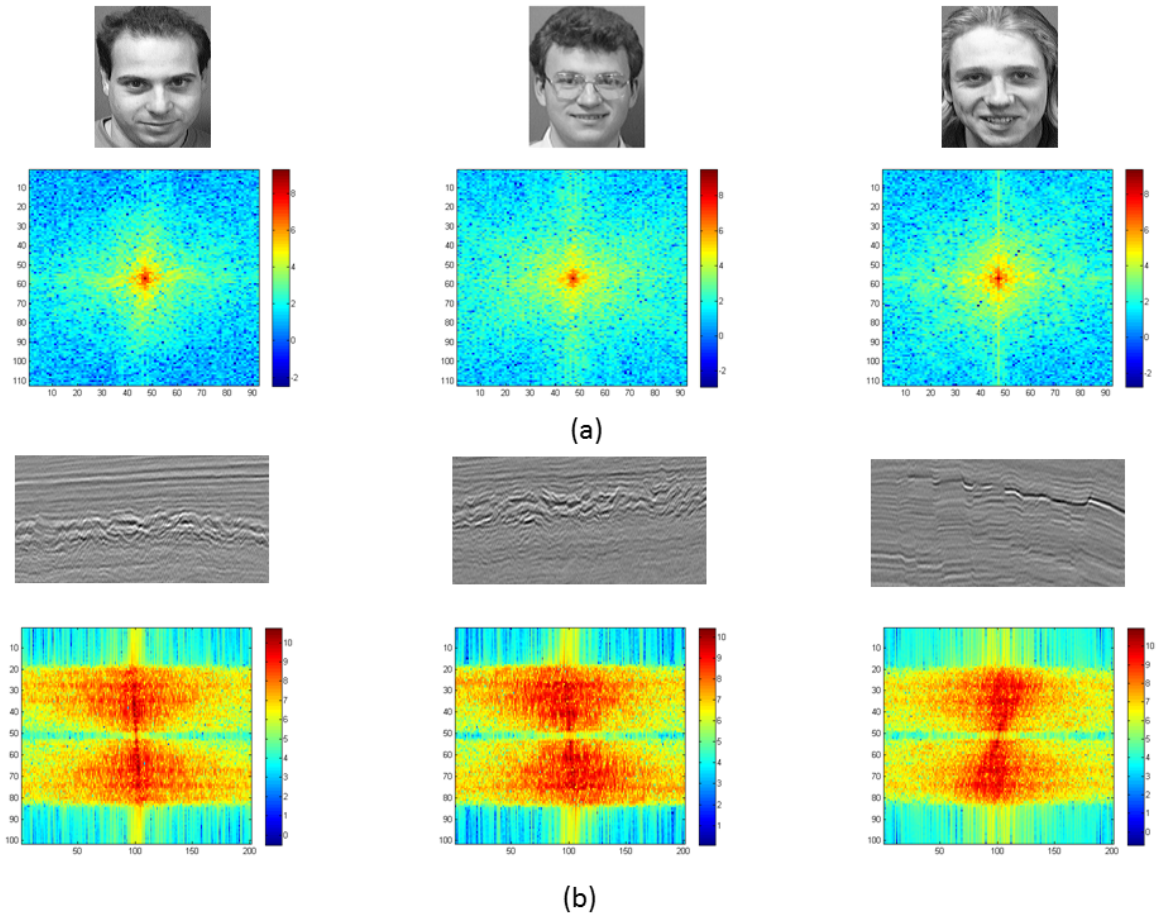


Figure 31: (a) Face images and their corresponding Fourier log-magnitudes plot (b) Seismic images and their corresponding Fourier log-magnitude plots.

3.10.1 Denoising Seismic Data

The developed training algorithm was tested on a seismic data set. Training data used was composed from $K = 25$ collections of seismic traces. Each collection is of size 550×100 data points. The curvelet transform is optimized using this training set. The training set was contaminated with AWGN with σ=1. Given the limited magnitudes of seismic traces in the training dataset, the value of σ chosen offers an adequate level of noise. When the value of σ is increased to two, the majority of seismic details were buried under noise. Varying the value of σ in a reasonable manner ($0.6 \leq \sigma \leq 1.4$) had a marginal impact on denoising results. The cost function used is denoising performance as measured by the log. of Mean Squared Error (MSE).

The number of scale decompositions J is found to be four for of all the images. The optimal scale locations are found by rounding the average K optimal results to the nearest integer. Similarly, optimal angular divisions were found by averaging the K optimal divisions and rounding to the nearest multiple of four.

Histograms for two inner scale locations Scale 1 and Scale 2 computed using the aforementioned training method are shown in Figure 32. The histograms show a clear preference for certain scale locations. The ranges of peak histogram counts values are well localized. For example, Most of V_1 counts fall between 480-500. Similarly, the majority of H_1 counts fall between 82-86. V_2 and H_2 peaks are within one to two coordinates from the default curvelet locations.

Next, Figure 33 shows histograms for the optimal choices of angular decomposition found for the same inner two levels. The angular counts show a behavior similar to scale location. The counts data are well localized and clearly show a preference for certain numbers of angular decompositions. It is worth noting that even when the counts spread over two or three different values, the count bins are adjacent to each other.

The computed trained adaptive curvelet tiles can be described by vectors

$$S_{adaptive} = \begin{pmatrix} 397 & 486 & 547 & 732 \\ 72 & 85 & 99 & 130 \end{pmatrix} \tag{19}$$

and

$$A_{adaptive} = \begin{pmatrix} 8 & 20 & 4 \\ 8 & 12 & 8 \end{pmatrix}, \tag{20}$$

where the first row of S indicates scale locations in the first quadrant. Similarly, the second row references scale locations in the second quadrant. Curvelet quadrants are illustrated in Figure 34.

Matrix A indicates the number of angular divisions used in each curvelet scale/quadrant

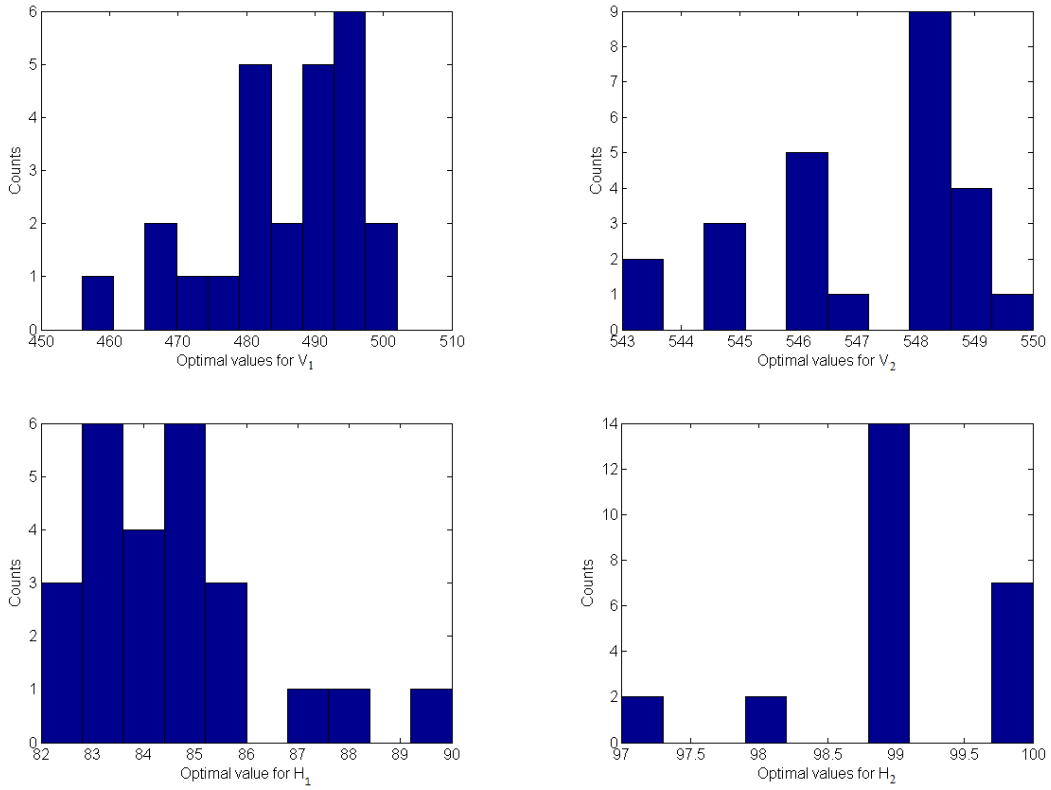


Figure 32: Histograms for computed optimal values for scale locations V_1, V_2, H_1, and H_2.

pair. The first and second rows correspond to the divisions used in the first and second curvelet quadrants respectively. Matrix A has fewer entries than S because of the non-directionality of the first curvelet scale. For comparison, the default curvelet scale locations are

$$S_{default} = \begin{pmatrix} 413 & 458 & 550 & 733 \\ 75 & 83 & 100 & 133 \end{pmatrix} \tag{21}$$

and the default angular decompositions per scale/quadrant pair are

$$A_{default} = \begin{pmatrix} 4 & 8 & 8 \\ 4 & 8 & 8 \end{pmatrix}. \tag{22}$$

In Figure 35, default and adaptive scale locations are plotted. The background of these images include a log-magnitude plot of the Fourier transform of a sample seismic image. Overall, adaptive curvelet scale locations are more uniformly distributed

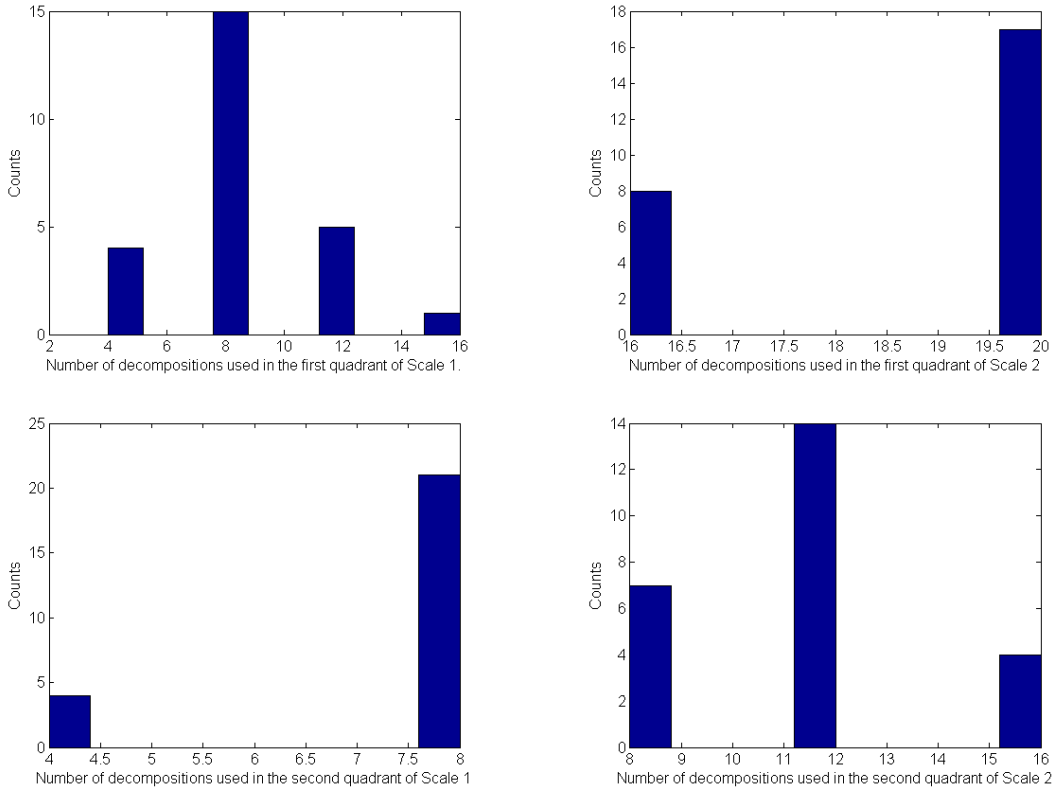


Figure 33: Histograms for computed optimal angular decompositions in Scale 1 and Scale 2.

across the Fourier plane. A major difference between default and adaptive scale locations is the location of scale location V_1. The value of this index increases from 458 to 486. This increase adds more space to the area covered by the first scale at the expense of a reduction in the area covered by the second scale. The inner most scale is also reduced in size. The value of V_0 decreases from 413 to 397, and the value of H_0 decreases from 75 to 72. This reduction also increases the size of Scale 1. The

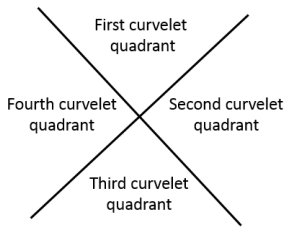
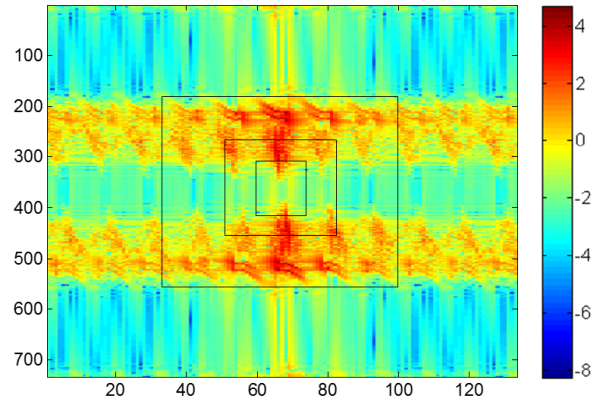
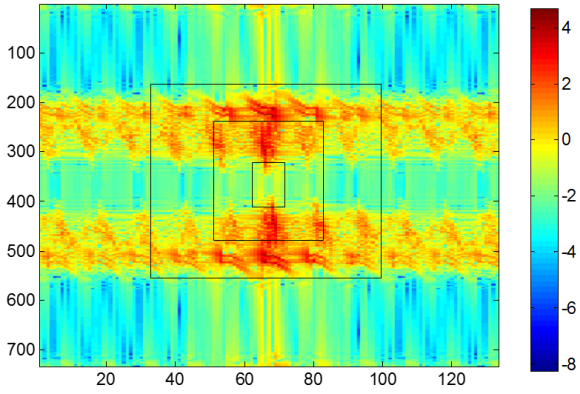


Figure 34: Curvelet quadrants.



(a) Default curvelet scale locations



(b) Adaptive curvelet scale locations

Figure 35: Comparison between default and adaptive curvelet scale locations. Scale locations are plotted on a log-magnitudes plot of the Fourier transform of a sample seismic image.

reduction in the size of inner most non-directional region, increases the directionality of the transform.

The number of angular decompositions in adaptive curvelets show a correlation with the signal's activity. This is most obvious with the distribution of angular divisions in the outer scale. When compared to other scales, Fourier transform activity in the most outer scale (Scale 3) is marginal. Adaptive curvelets reduce the number of divisions used in the outer scale from 16 to 12. H_3 has a higher number of decompositions than V_3. This can be explained by the activities observed to the right and the left of the Fourier plane. Such activity is not observed in the upper or lower part

of the plane.

The angular decomposition used in Scale 2 are higher than other scales. Close to 50% of the total number of angular divisions are used to describe this scale. This is a sign of the relative importance of the Fourier activity in this region. It is likely to play an important factor in the appearance of the overall image. The number of angular decompositions in Scale 1 is increased from four in default curvelets to eight. This increase indicates a need to increase the directionality of Scale 1 over what is proposed by default curvelets. The increase is associated with the increase in the size of Scale 1.

Adaptive curvelets increased the number of angular divisions used in all scales. The overall number of angular decompositions in adaptive curvelets was 60, while in default curvelets the number is 40 divisions. Adaptive curvelets increase the number of angular decompositions by 50%. The increase was not arbitrary and was related to the signal's Fourier transform activity. The number of angular decompositions in the outer most scale decreased. As mentioned earlier, this later decrease is related to the marginal level of Fourier transform activity in this scale.

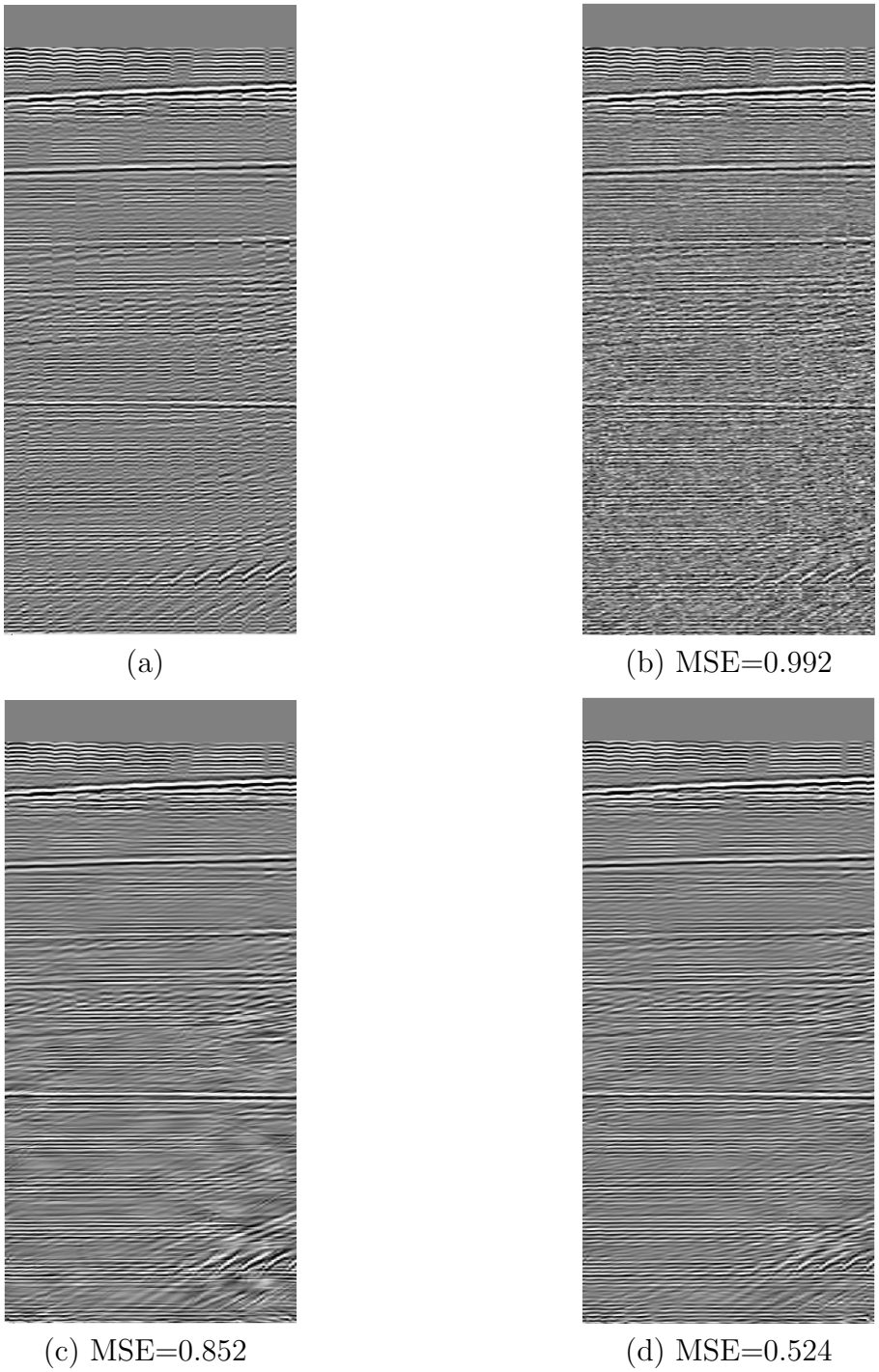
The optimal tilings found were applied to testing data and denoising results were compared with the original curvelet algorithm. Results show considerable quantitative and visual improvement as can be seen in Figure 36. Reconstructed image using adaptive curvelets show a reduction of mean squared error value (MSE) by 40%. The image reconstruted using adaptive curvelets have lower visual artifacts. This can be seen in the lower right segment of image Figure 36(d). The same region is corrupted by more artifacts in Figure 36(c), generated by default curvelets. Another visual difference between images Figure 36(c) and Figure 36(d) can be observed in the center of the image.

Comparison with the popular Non-Local Means (NLM) filter [10, 9] is shown in Figure 37. In this figure, adaptive curvelets outperforms default curvelets across all

noise levels σ. Default curvelets mean squared error increase in an exponential-like manner. While adaptive curvelets performance remains robust. NLM results were obtained using the author's recommended parameters. NLM denoising running time is about 10.4 minutes. Each Curvelet denoising result is obtained in less than three seconds. The adaptive tilings computed in this section, will be used to solve the problem of sparse recovery of seismic data in Chapter 4.

3.10.2 Denoising Face Images

Next, the algorithm is tested on face images using the AT&T Cambridge face database [54]. The database contains 400 face images taken from 40 subjects in 10 different facial expressions and positions. These images are of size 112×92. Two hundred images were used as a training set for the proposed curvelet transform. The remaining images were used as testing cases. Denoising performance was measured using PSNR and the Structural SIMilarity index (SSIM) [63]. SSIM values range from -1 to 1 and higher values indicate improved similarity. Figure 38 and Figure 39 show the results obtained for the testing images using $\sigma = 15$. Even though adaptive curvelets results are numerically close to default curvelets, visual improvements in performance are noticeable. This can be seen from the artifacts visible in Figure 40(b) shown in page 47. Four noisy image subjects with denoising results are shown in this figure. The adaptive tiling algorithm is compared with default curvelets and the non-local means filter. All adaptive curvelets results are free from the additive noise pattern. Curvelet images are faded when compared to the original image. This effect can be reduced by decreasing the value of k used in the thresholding algorithm (Section 3.1).



(a) (b) MSE=0.992

(c) MSE=0.852 (d) MSE=0.524

Figure 36: Denoising results (a)-(b) Original and noisy images, (c) Curvelet denoising (d) Adaptive curvelet denoising.

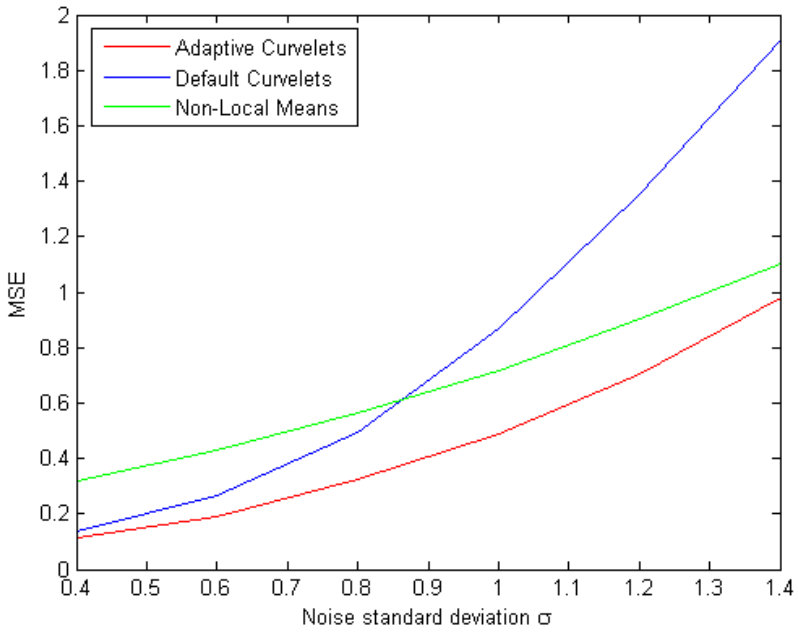


Figure 37: Seismic denoising results for default curvelets, adaptive curvelets, and NLM using various values of σ.

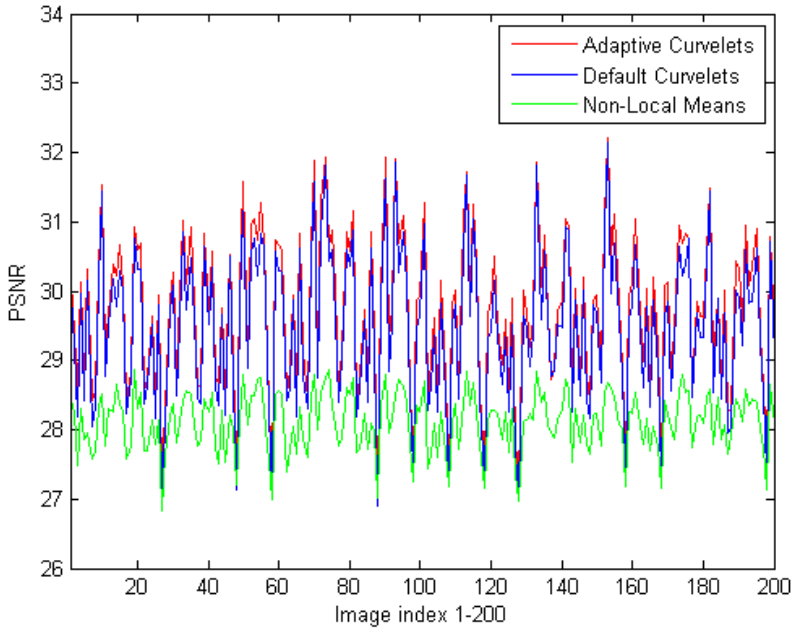


Figure 38: Denoising results (PSNR) for the AT&T face database.

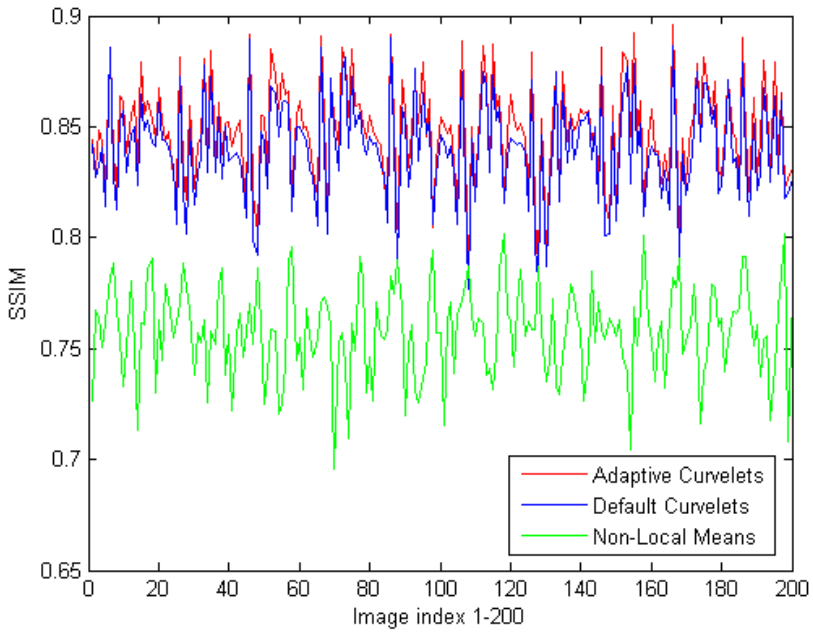


Figure 39: Denoising results (SSIM) for the AT&T face database.

Figure 40: Denoising results for the AT&T face database from left to right: (a) noisy image, restoration using: (b) default curvelets (c) NLM (d) adaptive curvelets.

CHAPTER IV

SPARSE DATA RECOVERY FROM SUBSAMPLED MEASUREMENTS

A signal is called sparse if most of its components are zero or close to zero. Most observed signals have a sparse representation that can be obtained using a sparsity inducing transform. The wavelet transform and its variants are examples of such transforms. A compressive sensing framework can recover sparse signals from subsampled measurements by solving a convex minimization problem. In this chapter, we present examples showing the advantages of using adaptive curvelets in the sparse recovery of subsampled data. A brief overview of compressed sensing theory is provided in the next section. This is followed by applications of adaptive curvelets in seismic data reconstruction from a subset of its traces. The chapter concludes by experiments on the sparse reconstruction of face images.

4.1 Background

Compressed sensing is a method for solving the problem of recovery of signals from a limited number of observations. It ensures that sparse signals acquired using appropriate measurement matrices can be recovered with high probability. The solution is computed by solving an l_1 norm minimization problem given by

$$\min \|\mathbf{x}\|_1 \quad \text{subject to} \quad \mathbf{y} = \mathbf{A}\mathbf{x}, \tag{23}$$

where $\mathbf{x} \in \mathbf{R}^N$ is the desired signal of length N. $\mathbf{y} \in \mathbf{R}^n$, where $n \ll N$, is the vector of observed measurements. $\mathbf{A}$ is the acquisition matrix of size $n \times N$. Numerous algorithms are available that solve the above optimization problem. The solution of (23) is guaranteed to recover the original signal $\mathbf{x}$ under conditions related to the sparsity of

$\mathbf{x}$ and the construction the acquisition matrix $\mathbf{A}$. Choices for $\mathbf{A}$ include partial Fourier measurements [15], realizations of white Gaussian noise [15], and Toeplitz matrices [6]. An incomplete identity matrix that randomly selects n out of N measurements is used as a choice for $\mathbf{A}$ in this dissertation.

Most signals are not sparse in the spatial domain. An inverse of a sparsity inducing transform is typically used prior to the application of the sensing matrix. Using the inverse curvelet transform C^{-1} in (23) yields the following equation:

$$\min \|\mathbf{x}\|_1 \quad \text{subject to} \quad \mathbf{y} = \mathbf{A}C^{-1}(\mathbf{x}). \tag{24}$$

In the new formulation, $\mathbf{x}$ becomes a vector representing the data in the curvelet domain.

4.2 Seismic Data Acquisition

Seismic reflection is an essential method for exploring subsurface layers of the Earth. It plays an important rule in the hydrocarbon industry, where it is used to locate and analyze hydrocarbon reservoirs. Acquisition of seismic data starts by generating seismic waves using a controlled active source of energy such as a dynamite explosion. The generated seismic waves interact with the subsurface layers of the Earth and generate reflections that are recorded using geophones at the surface. Each recorded wave is known as a trace. The recorded wave is processed using a structured collection of signal processing and mathematical tools to generate a map representing subsurface structure in the region of interest.

Sparsity was used in [32] to recover missing seismic traces from a subset of traces. The default curvelet transform was used as the sparsifying transform. Curvelet-based data recovery was shown to outperform wavelet and Fourier based recoveries. In what follows, improved recovery performance by using the adaptive curvelet transform is reported.

The problem of seismic data recovery from subsampled measurements can be mathematically described as follows. Assume that the data collected are in the form of vector $\mathbf{b} = \mathbf{Af} \in R^n$, where $\mathbf{f} \in R^N$ is the vector of complete data points and $\mathbf{A}$ is the sampling matrix that randomly samples $\delta = \frac{n}{N}$ seismic traces. Let $\mathbf{x}$ be the unknown curvelet coefficients vector representing $\mathbf{f}$. The vector of complete measurements can be approximated by solving the following program:

$$\hat{\mathbf{f}} = C^{-1}(\hat{\mathbf{x}}) \quad \text{with } \hat{\mathbf{x}} = \arg_{\mathbf{x}} \min \|\mathbf{x}\|_1 \text{ subject to } \mathbf{A}C^{-1}(\mathbf{x}) = \mathbf{b}, \tag{25}$$

where $C^{-1}(\mathbf{x})$ is the inverse curvelet transform of $\mathbf{x}$. It reshapes the 1D vector $\mathbf{x}$ into a standard curvelet coefficients format. Next, it takes the inverse curvelet transform and reshapes the constructed image into a 1D vector. Numerous algorithms that solve the above optimization problem are available . We used the Spectral Projected Gradient algorithm (SPGL1) [62, 33] for its efficiency and to facilitate comparisons with previously published results.

The operation of SPGL1 requires functions that resembles the sampling operation $\mathbf{A}C^{-1}(\mathbf{x})$ and its adjoint. The sampling matrix $\mathbf{A}$ can be very large in size. For example, the sampling matrix for a 512×512 image is of size $n \times 512^2$, where n is the number of sampled traces. This causes memory problems in modern computers. This issue is bypassed by introducing sampling functions that represents the operation of $\mathbf{A}C^{-1}(\mathbf{x})$ in an algorithmic manner.

The sampling operation $\mathbf{A}C^{-1}(\mathbf{x})$ takes the 1D vector $\mathbf{x}$ as a vector of data points representing the curvelet transform of the reconstructed image as input. It works by taking the inverse curvelet transform of $\mathbf{x}$, after transforming it into a regular curvelet representation (where the coefficients are organized to represent curvelet tiles). Next, the image is reshaped into a 1D vector and n traces are sampled. The adjoint operation takes the vector representing the sampled data points $\mathbf{b}$ and returns vector $\mathbf{x}$ the curvelet representation of the reconstructed image. It works by reversing the sampling operation. Values of zero were inserted for missing seismic traces. Vectors

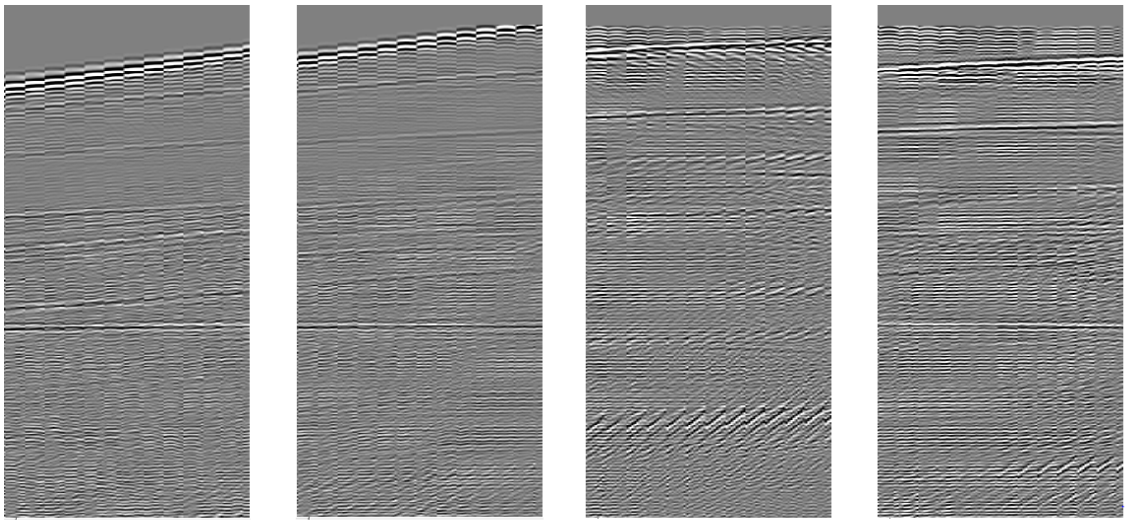


Figure 41: Four seismic images used in sparsity based reconstruction experiments.

b and **x** keep changing until convergence of the algorithm.

Equation (25) was solved using default curvelets and adaptive curvelets. Adaptive curvelet tiles were learned using the procedure described in Section 3.10.1, where 25 seismic data sets of size 550×100 were contaminated with AWGN to find the optimal adaptive curvelet tiles. Reconstruction performance, using a sampling ratio δ=0.3, is demonstrated using four different seismic data sets shown in Figure 41. Table 2 presents the mean squared error (MSE) values between original and reconstructed images obtained using default and adaptive curvelets. Adaptive curvelets succeed in generating recovery improvements in all cases.

Table 2: Sparse recovery results shown in MSE for four seismic data sets using a sampling ratio δ=0.3.

	Default Curvelets	Adaptive Curvelets
`Img1`	20.5	**15.5**
`Img2`	38.0	**26.1**
`Img3`	7.62	**6.11**
`Img4`	5.83	**3.63**

A Comparison of adaptive and default curvelet recovery performance for different values of δ is shown in Figure 42. The figure was generated using `Img4`. The

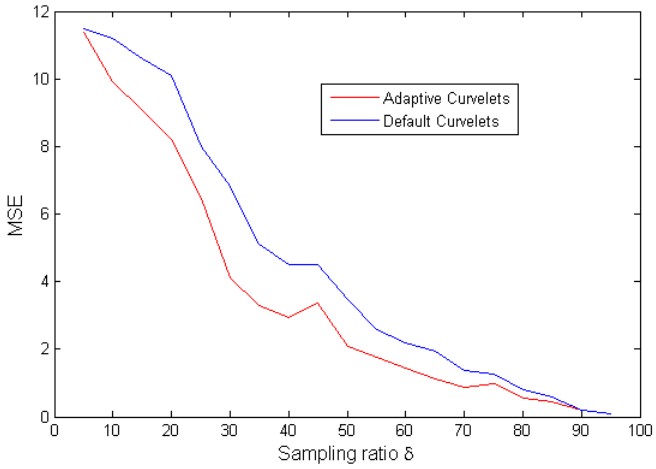


Figure 42: `Img4` recovery results across different subsampling ratios.

figure shows adaptive curvelets outperforming default curvelets over all subsampling ratios. Another example illustrating the performance of adaptive curvelet is shown in Figure 43. Figure 43(c) which is generated using adaptive curvelets, has lower artifacts and is closer in appearance to the original image. The artifacts visible in Figure 43(b) are in the form of spots of dotted discontinuities. These discontinuities are concentrated at the columns that were not subsampled. The dotted patterns are significantly reduced using adaptive curvelets.

4.3 Reconstruction of Face Images

In this section, adaptive curvelets are used in the sparse recovery of normal images from subsampled measurements. The AT&T face image database [54] was used in image recovery experiments. Adaptive curvelet tiles were learned using the procedure described earlier in Section 3.10.2. A total of two hundred were used as a training data set. The set includes face images of 20 subjects taken at 10 different settings (*e.g.* open or closed eyes and different poses). All images were of size 112×92. Adaptive curvelet tiles were learned using a noise standard deviation σ value equal to 15. Recovery results were demonstrated using a testing set of 52 images. These images were not part of the training data set.

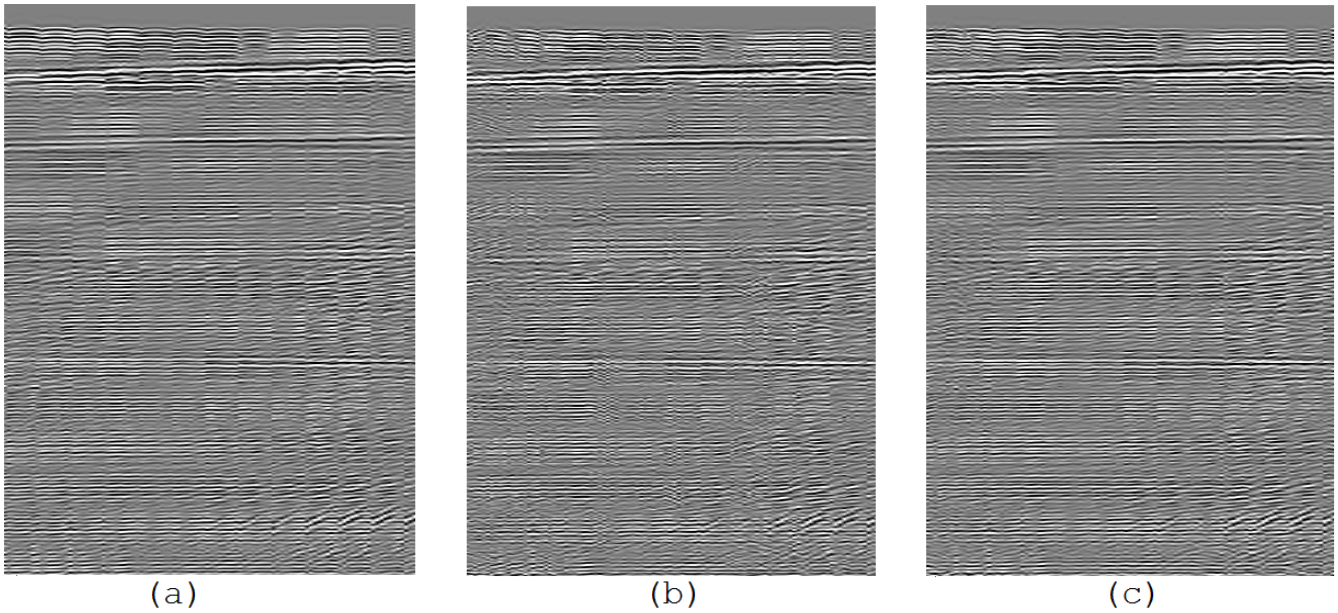


Figure 43: `Img4` recovery using $\delta = 0.6$. (a) Original image. (b) Recovery using default curvelets (MSE=2.47) (c) Recovery using adaptive curvelets (MSE=1.12).

Adaptive curvelets improved the recovery performance of default curvelets. Using a subsampling ratio $\delta = 0.4$, the average improvement in PSNR is 0.42dB. A histogram plot of the improvements achieved by using adaptive curvelets is shown in Figure 44. The improvements achieved by adaptive curvelets are established in the figure, however these improvements occupy a wide range with distinct peaks covering the interval from -0.1dB to 0.9dB. This behaviour is likely caused by the non-uniformity of the dataset used. The performance of adaptive curvelets can be further improved by training over face images exhibiting a higher degree of similarity. For example, the training and testing dataset can be selected to be from the same face setting in the AT&T database. Images of subjects wearing sunglasses or with beards can be trained separately. Three sample images along with recovery results are shown in Figure 45. In the first image, recovery using adaptive curvelets produces a sharper image that is less faded. For the second example, the image generated using adaptive curvelets show less artifacts. Details of the eyes and the region surrounding it are preserved better in the second subject's adaptive curvelet image. In the third image,

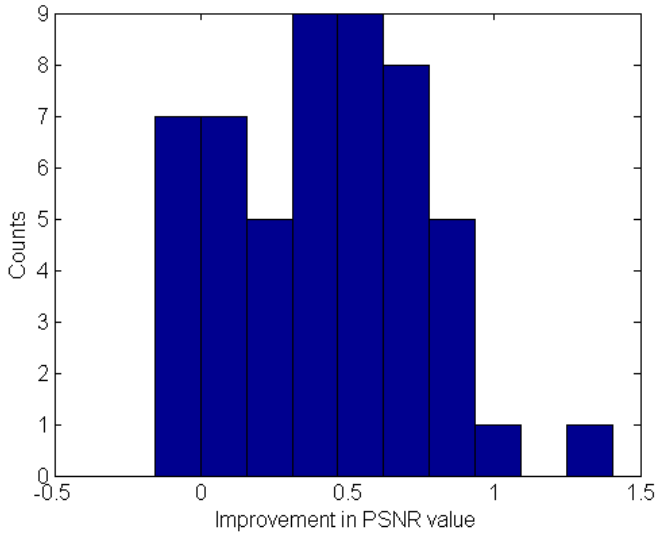


Figure 44: Histogram plot of adaptive curvelet denoising improvement (PSNR) over default curvelets.

reconstruction using adaptive curvelets generates less artifacts in the subject's right ear, mouth and nose regions. Overall, the appearance of adaptive curvelet images are more resembling of the original image, than the images reconstructed using default curvelets. Reconstruction performance in PSNR are shown in Table 3.

Table 3: Sparse recovery results in PSNR for the face images used in this study, δ=0.4.

	Default Curvelets	Adaptive Curvelets
`Image (a)`	26.59	**27.78**
`Image (b)`	31.07	**31.25**
`Image (c)`	31.19	**31.80**

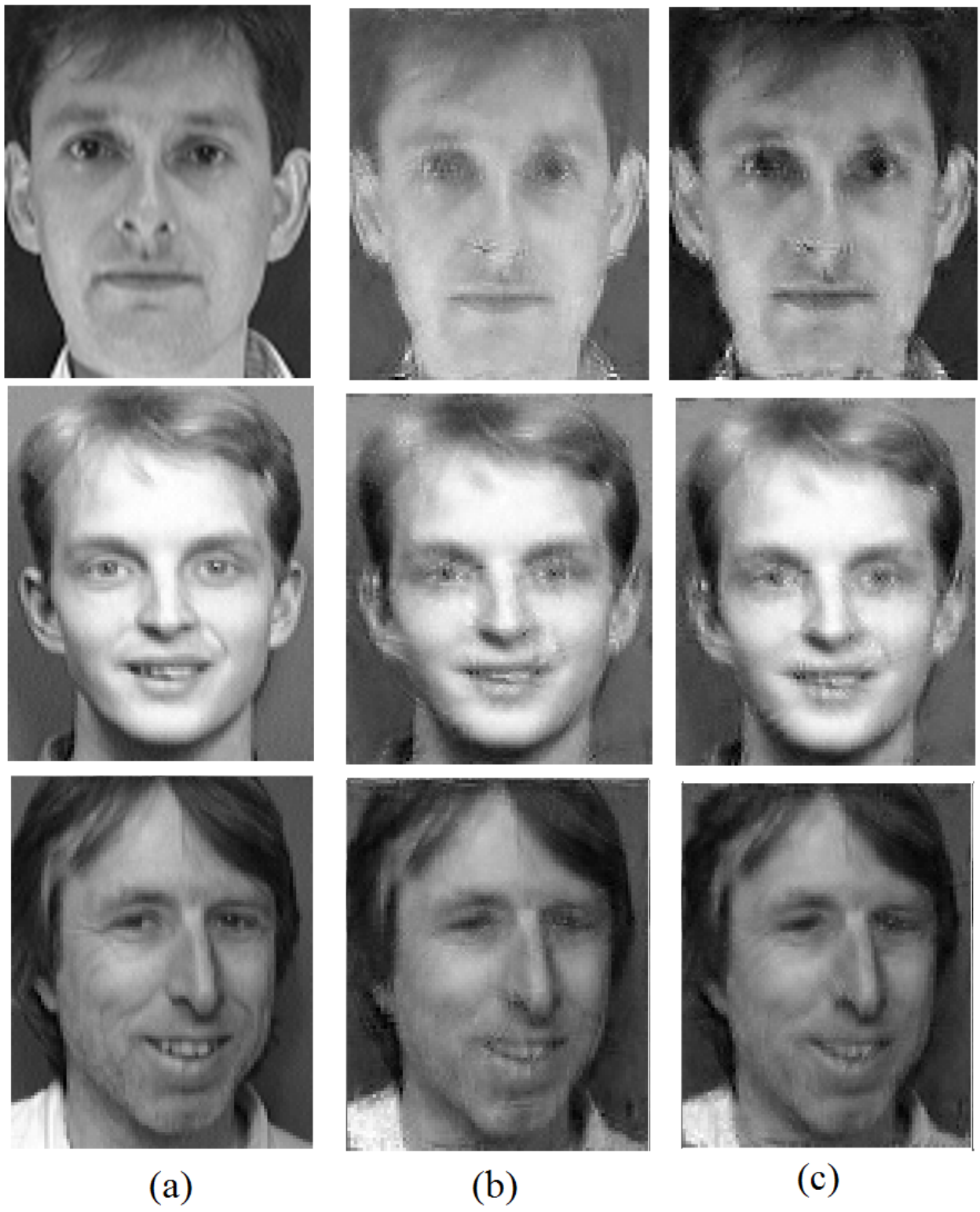


Figure 45: Visual comparison of sparse recovery performance ($\delta = 0.4$) between default and adaptive curvelets (a) Original image. (b) Recovery using default curvelets (c) Recovery using adaptive curvelets.

CHAPTER V

IMAGE SIMILARITY INDEX

Large amount of visual content is being generated and distributed at every instant. Productive use of such visual content demand the use of improved procedures for indexing, analysis, and classification of such material. These procedures rely on the visual-content of image/video databases. Textures are important components in images. Many algorithms have been proposed in the last decades that aim at developing content-based texture similarity measures. These algorithms are used in classifying and retrieving images of textured materials. They can be extended for use as elements in the framework of general content-based image retrieval systems. Curvelets without periodic extension were used in the literature to classify texture activities [65, 30]. In this chapter, texture classification algorithms based on periodically extended and adaptive curvelets are presented.

5.1 Introduction

Texture images exhibit a repeated pattern of visual content. Such repetitions can be captured using frequency domain techniques. The global holistic nature of such methods make them appropriate for texture analysis. Frequency domain texture retrieval algorithms are typically composed of two essential components: 1) a sparsity inducing transform that divides the spatial content of images into sets of coefficients representing unique sub-bands 2) a similarity measure that computes a numerical distance between image representations.

Do and Vetterli [25] proposed using the Kullback-Leibler distance between generalized Gaussian density (GGD) estimates of wavelet subbands. Several variants and modifications of algorithms using wavelets for texture retrieval were proposed in

Figure 46: Three Gabor basis elements at different orientations.

Figure 47: Examples of Curvelet basis elements.

the literature [37, 27, 64]. The wavelet transform offers limited directional selectivity (Wavelet sub-bands contain either horizontal, vertical, or diagonal information). Directional transforms allow more directional selectivity. They have been used successfully in a variety of application areas. Steerable pyramids [57], Gabor wavelets [39], and curvelets [13] are examples of directional transforms that have been used for texture retrieval. Zujovic et. al. [67] use statistical properties of a steerable pyramid representation of the texture data set. The statistical properties used include: mean, variance, horizontal and vertical autocorrelation, and crossband correlations. Gabor wavelets incorporates directional selectivity by using sets of Gaussian shaped filters at different rotations. Manjunath et. al. [45] use the mean and standard deviation of Gabor coefficients for texture retrieval. This Gabor-based algorithm is included in the multimedia content description standard MPEG-7 [52]. Gabor wavelets decomposes the input image into a set of Gabor elements. Figure 46 shows exemplary Gabor elements representing three different orientations. Recently, Zhang et. al. [65] reported performance improvements over Gabor filters by using the default curvelet transform. Curvelet elements are elongated needle-shaped elements that are generated by taking the inverse Fourier transform of anisotropic frequency bands, which

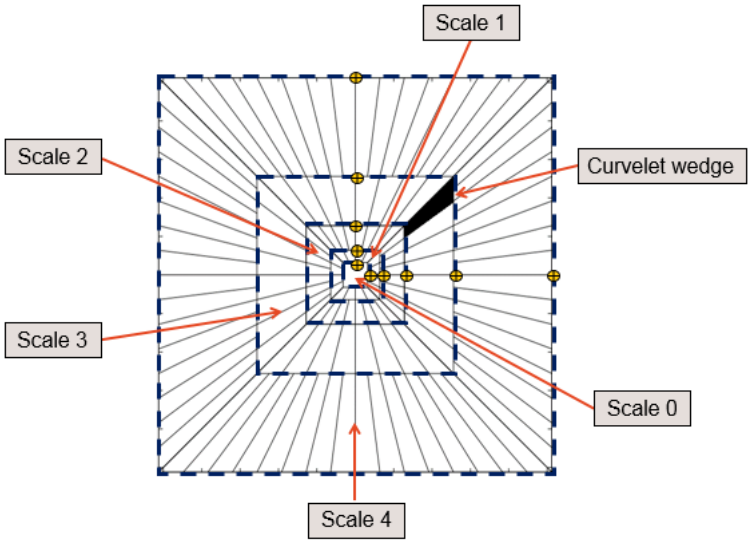


Figure 48: Default curvelet tiling. Scale locations are denoted by yellow markers.

we refer to as curvelet wedges. The length of each curvelet wedge is constructed to equal the square of its width. Figure 47 shows a number of Curvelet basis elements. Default curvelet tiling of the frequency domain is shown in Figure 48.

Finally, a number of authors developed texture retrieval algorithms based on spatial domain content analysis. Ojala et. al. [49] developed Linear Binary Patterns (LBP) that decompose input images into sets of coefficients representing intensity differences between a reference pixel and its neighbors. Local Tetra Patterns (LTP) [47] uses first order horizontal and vertical derivatives to represent the relationship between referenced pixels and their neighbors.

In this chapter, we propose new texture retrieval algorithms based on periodically extended and adaptive curvelets. Periodic extension improves default curvelet performance by connecting boundary elements. Adaptive curvelets vary the direction and thickness of curvelet basis elements to better represent image features.

5.2 Feature Vector and Distance Computation

Forming a representative feature vector is essential for the success of transform domain retrieval methods. Using the curvelet transform, the query and input images are represented by sets of coefficients identifying the spatial content relative to unique frequency bands. Curvelet elements are represented in the spatial domain by needle-shaped elongated objects as can be seen in Figure 47.

The similarity between two curvelet wedges representing a certain frequency band is indicative of the similarity of the spatial content in this band between the two images. Therefore, a global similarity measure between two images can be formed by including the similarities between curvelet wedges across all frequency bands. Given the fact that edges are essential components in images, a curvelet-based similarity measure between curvelet wedges is perceptually meaningful. Furthermore, adapting curvelet coefficients to better represent the query image constructs a representation from curvelet objects that are more faithful in direction and scale to the features of the image of interest.

Given two images A and B, where A is the query image, we propose the following algorithms for texture retrieval. The first algorithm uses default curvelets with periodic extension. The second algorithm uses adaptive curvelets. The periodically extended curvelet and adaptive curvelets are used to compute the curvelet coefficients representing images A and B. In adaptive curvelets, the optimal curvelet tiling will be learned using the query image A. Using this approach, image A is contaminated with Additive White Gaussian Noise (AWGN) with standard deviation σ that is given by the following equation:

$$\sigma = \max(std(A), 0.05 \times MAXI), \tag{26}$$

where $std(A)$ is the standard deviation of image A intensities, and $MAXI$ is the maximum possible intensity value. The lower bound for σ values used in (26) is

found to improve the performance and robustness of the adaptation algorithm. The learned frequency domain tiling will be used to find the adaptive-curvelet coefficients representing images A and B.

The feature vectors describing images A and B are formed by the mean and the standard deviation of curvelet coefficients in each wedge. Let the number of curvelet wedges in the optimal representation be M. The feature vector representing images A and B is given by

$$F = \{\mu_1, \sigma_1, \mu_2, \sigma_2, ..., \mu_M, \sigma_M\}, \tag{27}$$

where μ_i is the mean of curvelet coefficient in wedge i and σ_i is the standard deviation of the coefficients in the same wedge.

Next, the distance between images A and B is computed by

$$D(A, B) = \|F(A) - F(B)\|_1, \tag{28}$$

where $\|x\|_1$ is the l_1 norm of vector x. The feature vectors can be weighted to (de-)emphasize certain scales and orientation of interest (*e.g.* coarse/fine or horizontal/vertical features). A general expression for a weighted feature vector F is given by

$$F = \{w_1 \times \mu_1, w_1 \times \sigma_1, w_2 \times \mu_2, w_2 \times \sigma_2, ..., w_M \times \mu_M, w_M \times \sigma_M\}, \tag{29}$$

where $\mathbf{w} \geq 0$ is a weights vector of length equal to the number of curvelet wedges M.

5.3 Coefficient Sorting and Approximate Rotation Invariance

In this section, the proposed texture retrieval algorithm is extended to handle rotations. Rotating an image, rotates its Fourier Transform (FT) magnitudes [11]. Fourier transform contents described by a certain curvelet wedge in an image, is going to be moved to another curvelet wedge in the curvelet representation of the rotated image.

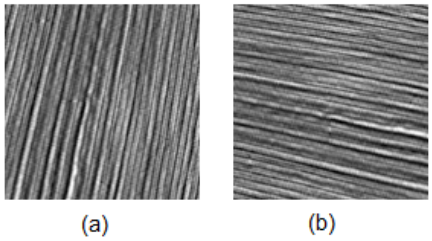


Figure 49: (a) Original image (b) Image 'a' rotated by 90°

By sorting curvelet wedges in each scale, one can reduce rotation artifacts in the distance computed between feature vectors representing two images. Curvelet wedges in each scale are sorted based on the sum of their coefficients magnitude. Once curvelet wedges are sorted, the distance between the images of interest can be computed. The distance will be based on feature vectors with aligned entries. This sorting mechanism was used successfully in [65].

Rotating the Fourier transform plane can also generate the undesired artifact of moving high magnitude FT data to lie across the boundary between two curvelet wedges. This in turn, makes the feature vector less representative of the similarity between the image and its rotated version. To reduce the effect of such cases, it is best to use a small value for the number of angular divisions. The number of angular decomposition for each curvelet scale/quadrant pair is set to four.

A texture image along with its 90° rotated variant are shown in Figure 49. The distance D between the two images using denoising based curvelets is 388.4. Using the rotation resilient measure, the distance D becomes 120.7.

5.4 Experiments

The proposed algorithm was tested in texture image retrieval using a variety of textured materials. The different quantitative measures that were used to assess the performance of the proposed algorithms are presented in the this section. This will

be followed by retrieval results from the datasets used in this study. To ensure robustness to various noise realizations, denoising-based adaptive curvelet results were obtained by using the average of five different trials.

5.4.1 Performance Metrics

Retrieval experiments are constructed using a query image representing a class of textured material. The proposed algorithms are assigned the task of acquiring images resembling the same class of the query image from a database of testing images. The following metrics are going to be used to evaluate the retrieval performance:

1. Retrieval precision: Defined as the ratio between the number of correct matches retrieved, to the total number of matches retrieved. Precision in this context is a function of the number of retrieved images. We will focus on precision values at one, two, and at the number of relevant images.

2. Mean reciprocal rank (MRR): It is the mean reciprocal of the rank of the first relevant position.

3. Mean Average Precision (MAP): It is the mean of average precision values across all queries. The average precision (AP) is given by

$$AP = \frac{\sum_{\forall k} P(k)}{N}, \tag{30}$$

   where k and N are the ranks and number of relevant images, respectively.

4. Precision-recall curves: These figures are frequently used in the analysis of retrieval performance. In such figures precision values are plotted as function of recall values. Recall is defined as the ratio between the number of correctly retrieved images and the number of relevant images.

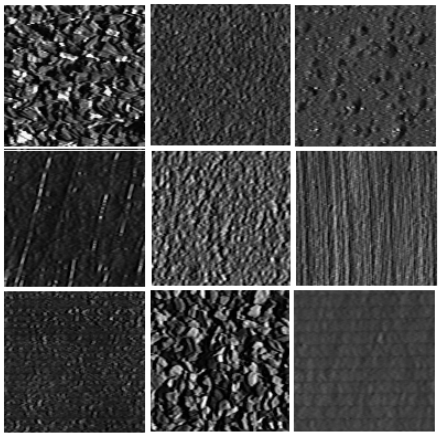


Figure 50: A sample from the CUReT texture database.

5.4.2 Texture Retrieval: Experimental Setup

The algorithm is tested using three different texture databases: the Columbia-Utrecht Reflectance and Texture Database (CUReT) [21], Mondial Marmi collection of granite classes [7], and the fabric dataset included in the Salzburg Texture Image Database (STex) [2].

The CUReT database contains 61 images of real-world surfaces taken at different illumination and viewing directions. In our experiments, we used images taken at the first illumination setting with viewing direction 22. CUReT images are of size 640 $\times$ 480. They all include a textural part and a background. Three images of size 128 $\times$ 128 covering the textural regions were extracted from each CUReT image. One of these images were selected randomly as a query image. The remaining images are used as testing images. A sample of the images used in this study is shown in Figure 50.

The STex dataset includes 476 images representing different types of materials. In our experiments, we use the fabric class of images in this dataset. It includes 77 fabric samples of size 512 $\times$512. Each image is divided into four non-overlapping regions. This makes the total number of images equal to 308. An image in each class is chosen as a query image. The remaining images are used as testing images. A sample of the

Figure 51: Samples from the STex fabric dataset.

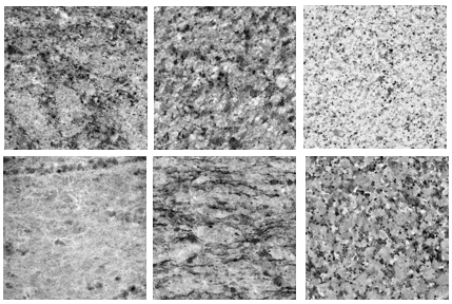


Figure 52: A sample from the Mondial Marmi texture database.

images used in this study is shown in Figure 51.

The Mondial Marmi is a collection of 48 images representing 12 different granite classes. Each image is hardware-rotated using nine rotation angles: 0°, 5°, 10°, 15°, 30°, 45°, 60°, 75° and 90°. The images have been acquired under controlled illumination conditions. Each image is of size 544×544. To save computational time, granite images were downsampled using a downsampling factor of 1/3. Query images included an image from each class along with its rotated versions. The remaining images were used as testing images. The number of correct matches for each query image is equal to 3×9. Samples from the Mondial Marmi data set are shown in Figure 52.

5.4.3 Texture Retrieval Results

The performance of the proposed methods is compared with the following algorithms:

1. l^2 norm of default curvelet features [65]. This algorithm uses no periodic extension or angular divisions in the outer curvelet scale.

2. l^1 norm of Gabor features [45].

3. Kullback-Leibler distance on wavelet features [25].

4. Log-likelihood ratio of Local Binary Patterns [49]. The used feature vector is composed from the sum of $LBP^{riu2}_{8,1}$ and $LBP^{riu2}_{24,3}$ features.

Implementations of the algorithms based on Gabor, wavelet, and LBP were downloaded from the authors websites. The results of experiments performed on the CUReT database are shown in Table 4.

Table 4: Retrieval results for the CUReT dataset.

	P@1	P@2	MRR	MAP
Gabor features	0.934	0.869	0.962	0.901
Wavelet features	0.967	0.861	0.980	0.924
Default curvelet	0.951	0.861	0.975	0.916
Linear Binary Patterns (LBP)	0.934	0.885	0.959	0.925
Curvelet with periodic extension	**1.000**	0.934	**1.000**	0.964
Adaptive curvelets	**1.000**	**0.939**	**1.000**	**0.968**

Precision at one results indicate that the proposed algorithms succeed in retrieving one of the correct matches as a first retrieved image. Precision at two (P@2) results show that the proposed methods are able to retrieve 93-94% of the correct matches in the database. The proposed methods outperform the other algorithms in MAP by a difference of at least 4%. Precision-recall curves for methods based on LBP, Gabor features, default curvelet, curvelet with periodic extension, denoising based adaptive curvelets are shown in Figure 53.

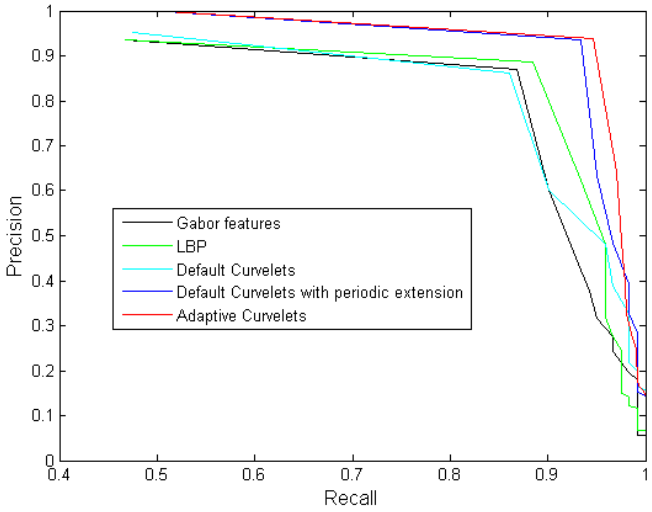


Figure 53: CUReT Precision-recall curves.

Similarly, results for experiments performed on the fabric database are summarized in Table 5.

Table 5: Retrieval results for the Fabric dataset.

	P@1	P@2	P@3	MRR	MAP
Gabor features	0.948	0.916	**0.887**	0.961	0.907
Linear Binary Patterns (LBP)	0.896	0.779	0.680	0.928	0.760
Wavelet features	0.935	0.844	0.792	0.959	0.842
Default curvelet	0.779	0.669	0.571	0.837	0.636
Curvelet with periodic extension	**0.974**	0.909	0.849	0.983	0.896
Adaptive curvelets	**0.974**	**0.927**	0.870	**0.984**	**0.918**

Curvelet based methods outperform other algorithms in the precision of the first retrieved result (P@1). Overall, adaptive curvelets bypasses other retrieval methods. Gabor features achieve results comparable to adaptive curvelets. The low performance of Gabor features in P@1 makes it unattractive for applications where the user is critical of the quality of the first retrieved image. Adaptive curvelet results can be further improved by training over a set of images resembling the class of interest. The training procedure as it is currently implemented uses the query image. Training on the query image alone can overemphasize its local features. Precision-recall curves for

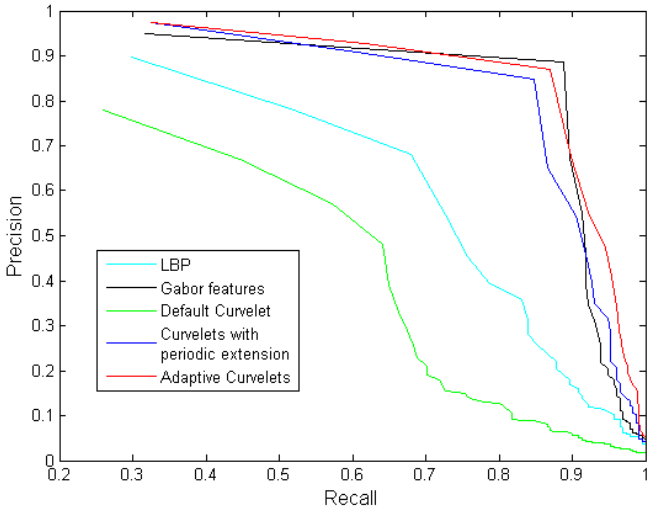


Figure 54: Precision-recall curves obtained for the fabric database.

methods based on wavelets, Gabor features, default curvelet, curvelet with periodic extension, denoising based adaptive curvelets are shown in Figure 54.

Finally, capabilities of the rotation-normalized version of the proposed algorithms are presented. Results for experiments performed on the Mondial Marmi granite database are shown in Table 6.

Table 6: Retrieval results for the Mondial Marmi granite database. Gabor and Curvelet based algorithms were normalized to rotations.

	P@1	P@2	P@27	MRR	MAP
Gabor features	0.898	0.903	0.796	0.937	0.826
Linear Binary Patterns (LBP)	0.972	0.954	0.831	0.973	0.880
Default curvelets	0.778	0.773	0.623	0.826	0.704
Default curvelet with periodic extension	0.982	0.977	0.857	0.989	0.907
Adaptive curvelets	**1**	**1**	**0.939**	**1**	**0.973**

Adaptive curvelets results are compared with results acquired using LBP and a rotation-normalized version of the gabor retrieval method. Gabor retrieval algorithm is normalized to rotations by using the approach detailed in Section 5.3. Local binary patterns are rotation-invariant by construction. The number of division per Gabor

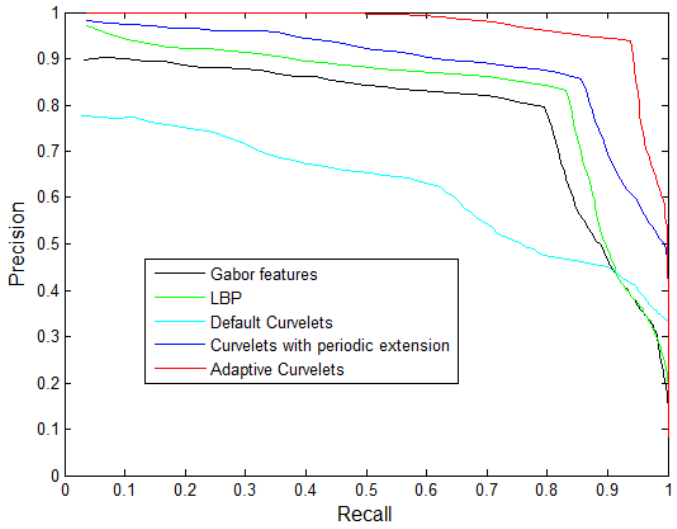


Figure 55: Precision-recall curves for the Mondial Marmi granite database.

scale/quadrant pair is kept at the default value of three. Table 6 presents the experimental results, where the performance of adaptive curvelets is compared with LBP and the gabor feature algorithms. The rotation normalized adaptive curvelets algorithm clearly outperforms the other algorithms. Adaptive curvelets achieves perfect retrieval precision up to a recall value of 0.5. Precision-recall curves are shown in Figure 55.

5.4.4 Seismic Activity Characterization

Seismic data are used extensively in the oil and gas exploration industries. Developing efficient retrieval algorithms based on visual properties of the seismic datasets is of crucial significance. Adaptive curvelets retrieval algorithm is tested on the seismic dataset shown in Figure 56. The figure covers the following seismic activities:

Fault Data sets with faults

Horizontal Data sets with obvious horizons

Dome Data sets with salt dome shapes and alike

Clear Data sets with none of the above activities

Table 7: Retrieval results for the seismic dataset

	P@1	P@2	MRR	MAP
Adaptive curvelets	0.75	0.75	0.85	0.83
Weighted adaptive curvelets	**0.92**	**0.83**	**0.93**	**0.91**

For efficient retrieval performance, a weighted version of the adaptive curvelet algorithm is used (Section 5.2). The number of curvelet scale decompositions is chosen, by the scale selection algorithm, to equal four. `Fault` images were retrieved using curvelet wedges representing vertical activities . These wedges lie in the second curvelet quadrant (Figure 57). Only, the finer (outer) two curvelet scales are used for retrieval. Weights for the inner scales were set to zero. Similarly, `Horiz` datasets were retrieved using curvelet wedges representing horizontal activities (first curvelet quadrant). The inner two curvelet scales (coarsest levels) were used for retrieval. `Dome` images were retrieved using the outer two curvelet scale and all curvelet directions. `Clear` images were retrieved using all curvelet scales and directions.

Adaptive curvelet tiles were found by maximizing the coefficient of variation (C_v), where $C_v = \frac{\sigma}{\mu}$, of curvelet coefficients magnitude. This cost function is found to improve the performance of default curvelets in seismic applications [3]. A closer look into this cost function is going to be provided in the next chapter.

Every image in the dataset was used once as a query image, where the remaining images were used as testing images. Comparison between unweighted and weighted versions of adaptive curvelets retrieval performance are shown in Table 7. Unweighted adaptive curvelets succeed in retrieving the first correct match in nine out of 12 images in the database. Weighted curvelets increase this ratio to 11 out of 12, where the algorithm matches image `Horiz3` with `Clear3` and `Clear1` instead of the correct `Horiz` images.

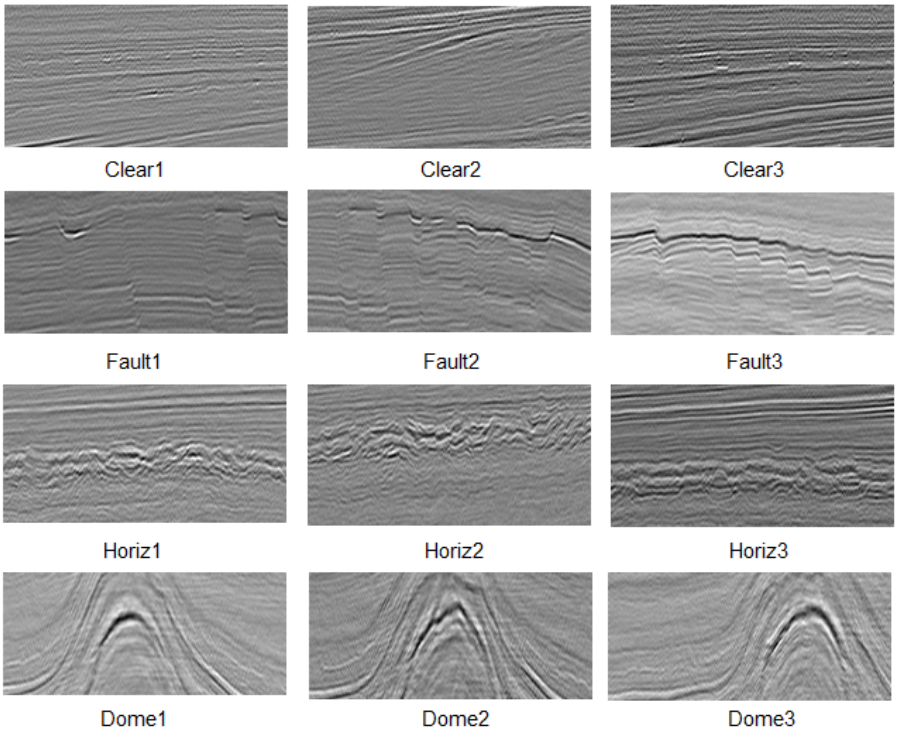


Figure 56: Seismic images representing different classes of seismic activities.

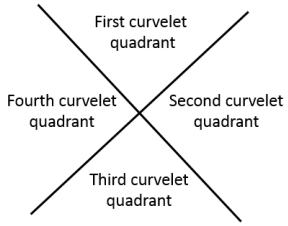


Figure 57: Curvelet quadrants.

5.5 Conclusions

New algorithms for texture image retrieval were presented in this chapter. Their performance was tested on various texture datasets. Our results strongly encourage the use of periodically extended curvelets despite the increase in computational cost. Adaptive curvelets were shown to improve default curvelet performance by adapting curvelet tiles to better represent the image of interest. Adaptive curvelets did also achieve promising results in retrieving seismic data. To reduce computational costs, training of adaptive curvelets can be performed prior to receiving image queries.

In this case, the search for optimal curvelet tiles is going to be performed over a set of images representing the class of interest. The texture database will consist of the images representing various material classes along with the optimal curvelet representation found per class. This work can be extended to video and other three dimensional datasets. The robustness of mean and standard deviation based feature vectors can be improved by normalizing their elements by the mean and the standard deviation of curvelet coefficients in the entire database. An interesting research path to follow is to incorporate the presented algorithms as a part of a general image retrieval system.

CHAPTER VI

ALTERNATE COST FUNCTION

An alternate cost function that substitutes denoising-based curvelet adaptation is proposed in this chapter. The new cost function is based on maximizing the coefficient of variation of adaptive curvelet coefficients. Experimentally, we have observed that this cost function achieve results comparable to denosiing-based adaptive curvelets in seismic datasets. It was also found to achieve better results in the CUReT database texture classification problem. The proposed cost function eliminates the need for a training dataset. It also achieves a reduction in computational cost. Seismic experiments run with different sparsity inducing cost functions are presented in the next section. This is followed by results obtained on the CUReT database. Statistical properties of seismic and CUReT curvelet representations are going to be presented next. The chapter concludes with remarks proposing ways to generalize the C_v based cost function to other datasets.

6.1 Inducing Sparsity in the Seismic Domain

Three sparsity inducing cost functions are tested. These functions are:

1. Minimizing Shannon's entropy. This measure was used by Coifman et. al. [19] to find the best wavelet basis in wavelet packets. It is given by

$$J = -\sum_i p_i \, log(p_i). \tag{31}$$

   The number of quantization bins used is 256.

2. Maximizing the Gini index. The Gini index is a measure of economic statistical dispersion that was recommended for use as a measure of data sparsity in [34].

It can be described as

$$J = 1 - 2\sum_{k=1}^{N} \frac{c_{(k)}}{||\mathbf{c}||_1}(\frac{N-k+1/2}{N}), \tag{32}$$

where N is the number of coefficients and $c_{(k)}$ is the k-th coefficient where the coefficients are sorted so that $c_1 \leq c_2 \leq ... \leq c_N$. Gini index ranges between zero and one. A zero Gini index value indicates a vector with equal entries and one indicates a vector with maximum sparsity.

3. Maximizing the coefficient of variation which is the ratio between the standard deviation of curvelet's coefficients vector and its mean value. Mathematically, the function can be expressed as

$$J = \frac{\sigma(|\mathbf{C}|)}{\mu(|\mathbf{C}|)}, \tag{33}$$

where $\mathbf{C}$ is the vector of curvelet coefficients. Equation (33) induces sparsity by setting the mean of curvelet coefficients as close as possible to zero, while maximizing the standard deviation σ of curvelet coefficients.

The three measures are tested on denoising five seismic data sets of size 550 $\times$ 100. The results obtained are summarized in Table 8. All proposed cost functions succeed in generating improvements in denoising performance over default curvelets. The coefficient of variation approach achieves the best results.

Table 8: Average denoising performance of default and adaptive tiling curvelets. Five seismic data sets of size 550 $\times$ 100 were used in this experiemnt.

	Default Curvelets	Adaptive Curvelets		
		Entropy	Gini index	C_v
Avg. MSE $\sigma = 1$	0.8230	0.4930	0.4934	**0.4710**
Avg. MSE $\sigma = 1.5$	1.3600	0.9196	0.9084	**0.8288**

The scale locations and angular decomposition vectors found by the C_v optimizer were very close to the ones generated by denoising based adaptive curvelets. When

compared to denoising-based adaptive curvelets, C_v based curvelets offer reductions in computation time by 40%-50%. This reduction can be explained by the lower computational cost of the coefficient of variation. In each iteration, C_v computation requires a forward curvelet transform operation. This is followed by low complexity mathematical computations. In comparison, denoising-based adaptive curvelets require computing the forward curvelet transform, thresholding, and the inverse curvelet transform. These operations are also followed by a PSNR calculation. Another advantage of the C_v based optimizer is that it does eliminate the need for training datasets.

6.2 CUReT Texture Classification

Next, we report results generated by the C_v adaptive curvelet in the CUReT texture classification problem that was presented earlier in Section 5.4.2. Table 9 summarizes retrieval results acquired using seven different algorithms. Details about these algorithms and the cost functions used were presented in Section 5.2 and Section 5.4.3. Precision at one results indicate that the proposed algorithms succeed in retrieving one of the correct matches as a first retrieved image. Precision at two (P@2) results show that the proposed adaptive curvelet methods are able to retrieve 93-96% of the correct matches in the database. The proposed methods outperform the other algorithms in MAP by a difference of at least 4%. Precision-recall curves for methods based on LBP, default curvelet, curvelet with periodic extension, denoising and C_v based adaptive curvelets are shown in Figure 58.

6.3 Coefficients Decay and Directions for Future Research

The success of C_v based adaptive curvelets with seismic datasets and with the CUReT database did not generalize to other examples. In this section, we present an analysis of images where the C_v cost function fails to achieve results comparable to denoising based adaptive curvelets. Curvelet coefficients of these images will be compared with

Table 9: Performance comparison of the different algorithms used in C_v based image retrieval.

	P@1	P@2	MRR	MAP
Gabor features	0.934	0.869	0.962	0.901
Wavelet features	0.967	0.861	0.980	0.924
Default curvelet	0.967	0.861	0.983	0.918
Linear Binary Patterns (LBP)	0.934	0.885	0.959	0.925
Curvelet with periodic extension	**1.000**	0.934	**1.000**	0.964
Denoising based adaptive curvelets	**1.000**	0.951	**1.000**	0.969
C_v based adaptive curvelets	**1.000**	**0.959**	**1.000**	**0.975**

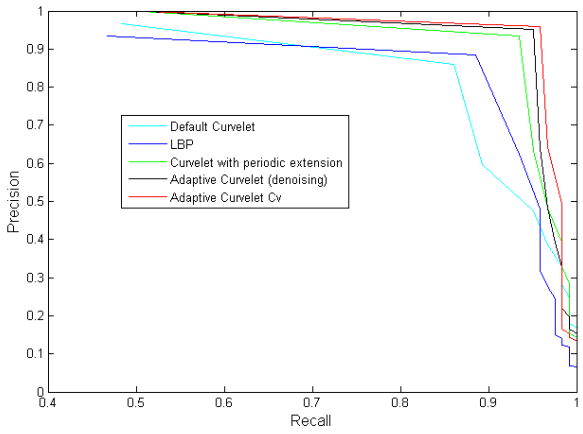
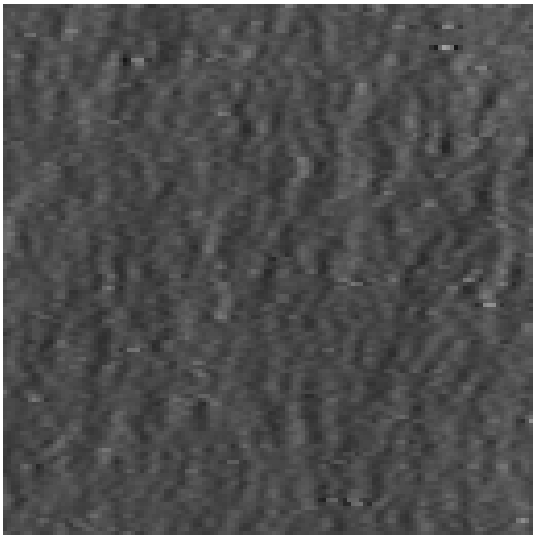


Figure 58: Precision-recall curves for five different algorithms used in C_v based image retrieval.

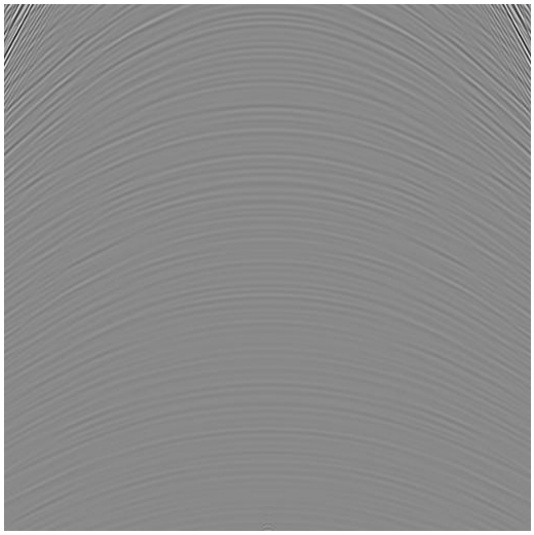
the curvelet representation of images where the C_v optimizer achieves results similar or better than denoising-based curvelets. The goal is to better understand this optimizer and to develop insights for future research generalizing the C_v cost function to work with general datasets.

In general images where the C_v optimizer works well have lower curvelet coefficients magnitudes than other images. A sample of the images used in our experiments are shown in Figure 59. All of these images are of size 256×256. The CUReT texture image generates the best performance in texture retrieval results using the C_v optimizer. The C_v optimizer generates suboptimal results with the fabric

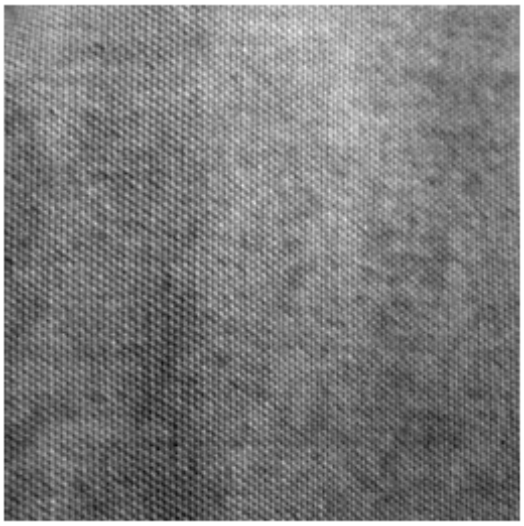
texture. In denoising experiments, image Barbara achieves optimal results with denoising based adaptive experiments. The C_v optimizer generates results worse than default curvelets. Adaptive curvelets based on C_v generate improvements over default curvelets in the seismic dataset shown in Figure 59(b).



(a) CUReT texture image



(b) Seismic image



(c) Fabric texture image

(d) Barbara

Figure 59: The C_v optimizer works well with the images (a) and (b). Suboptimal results are generated with images (c) and (d).

The behaviour of the C_v optimizer depends on the mean and standard deviation of curvelet coefficients. Table 10 presents the magnitudes of sorted curvelet coefficients

for the four images at different coefficients numbers. In these images, default curvelets generate about 480×10^3 coefficients. When compared with the Fabric image, The CUReT image has a reduction in the highest curvelet coefficient magnitude by a factor of 50%. About 40% reduction is noticed between the seismic and Barbara images. Curvelets coefficients for the CUReT and the seismic image exhibit sharper tail-decay behaviour. The decay of the coefficients is illustrated in Figure 60 where coefficients magnitudes for the four images are plotted on a log-log scale. Figure 61 plots coefficients magnitude as a function of the total number of coefficients. The figures shows that the tails of seismic and CUReT image coefficients decay faster.

Table 10: Default curvelet coefficients magnitudes at different coefficient numbers for images `CUReT`, `Seismic`, `Fabric`, and `Barbara`. Coefficient of variation optimization works well with the first two images and generates suboptimal results for the later two.

	1st coefficient	30×10^3-th coefficient	300×10^3-th coefficient
`CUReT`	1114.4	5.0	0.2
`Seismic`	1679.8	2.5	0.2
`Fabric`	2038.4	6.5	0.9
`Barbara`	2942.5	12.5	1.0

Future research is recommended to address adding a weighted variant of the coefficient of variation where the the highest curvelet coefficient does not overpower other coefficients. Ideas can be borrowed from the analysis of other long tailed problems. A closer look into convergence properties of the scale search algorithm is likely to contribute to the development of an improved C_v optimizer. This convergence analysis should study cases where the coefficient of variation optimizer converges to local minima.

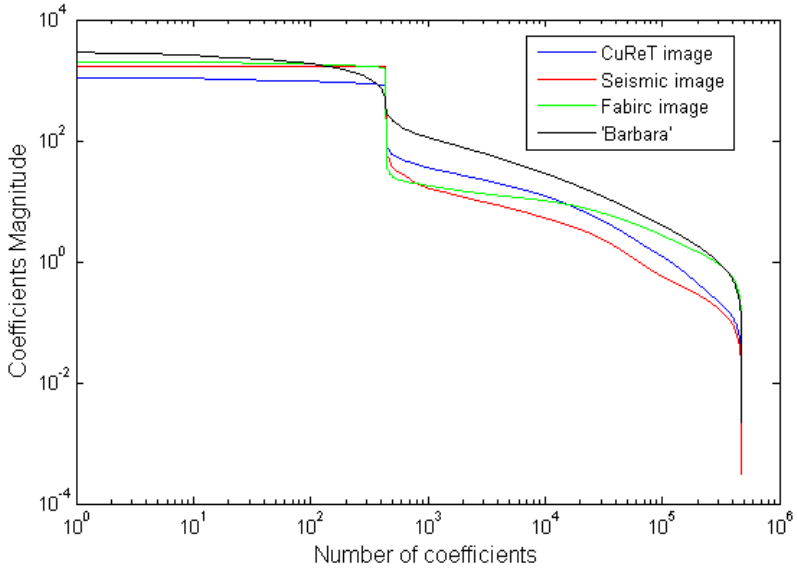


Figure 60: Decay of curvelet coefficients for images `CUReT`, `Seismic`, `Fabric`, and `Barbara`. The coefficients are plotted using a log-log scale.

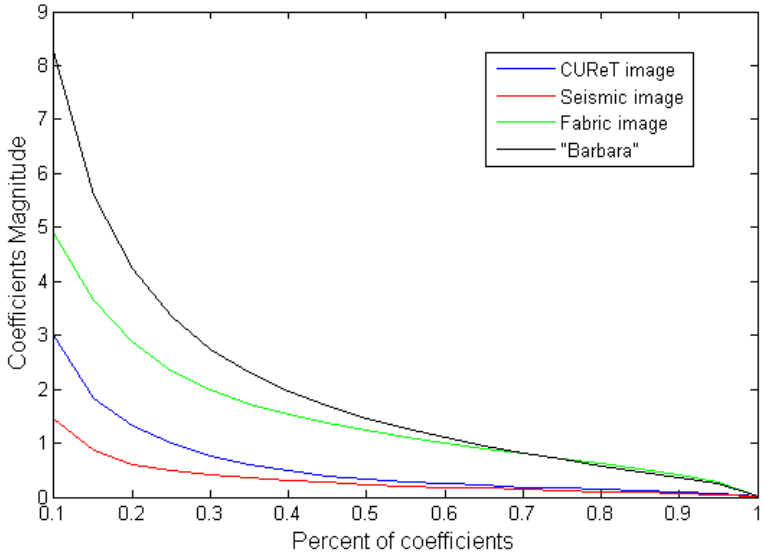


Figure 61: Decay of curvelet coefficients for images `CUReT`, `Seismic`, `Fabric`, and `Barbara`. The figures plots coefficients magnitudes as a function of the total number of coefficients.

CHAPTER VII

CONCLUSIONS

A new approach to designing wavelet and wavelet-like transforms is proposed in this dissertation. The method relies on adapting transform parameters until the maximum of a chosen cost function is achieved. The proposed method was designed using the curvelet transform as the basis algorithm. Curvelet's default scale locations, angular decompositions and number of scales were optimized to maximize the performance of a given cost function. The primary cost function used in this dissertation is the PSNR of an artificially generated noisy image that was constructed to be denoised using adaptive curvelets.

The performance of adaptive curvelets was demonstrated using a variety of application areas. Applications included in this dissertation include image denoising, and image reconstruction from a partial set of coefficients.

Next, adaptive curvelets were used in the sparse recovery of subsampled seismic datasets. Results demonstrate performance improvements over default curvelets. This problem is of importance to the seismic community where the need for enhanced seismic images is constantly evolving despite acquisition costs and physical constraints.

An image similarity index was constructed using adaptive curvelets. It achieved results outperforming other widely used algorithms. The algorithm was tested using the CUReT texture dataset, fabric textures from the STex dataset, and granite datasets. A weighted version of the similarity index was also proposed and used successfully in the classification of seismic activities.

Finally, an alternate cost function that depends on the coefficient of variation of

curvelet coefficients was proposed. The proposed cost function eliminates the need for training datasets. It also reduces computational time by a factor of 40%-50%. The proposed cost function did achieve results comparable and outperforming denoising-based adaptive curvelets in seismic applications. More research is needed to generalize the C_v based optimizer for use with general images.

The success of adaptive curvelets encourages transferring the approach used in this dissertation to other transforms. Examples of recently introduced wavelet-like transforms include shearlets, contourlets, and the easy path wavelet transform. Depending on the application and the transform used one can vary the adaptation parameters. For example, controulets use a time-domain implementation that is similar to a curvelet setup. The adaptive parameters in this case will be ones that are essential in such time-domain constructions.

An interesting research problem to study is the design of general free-form transforms where the signal's activity is broken down into different segments with minimal structural and transform dependent parameters.

REFERENCES

[1] "Curvelet toolbox version 2.1.2." `http://www.curvelet.org/software.html`, 2007.

[2] "Salzburg texture image database (STex)." `http://wavelab.at/sources/STex/`, 2009. [Online; accessed 26-Sep-2014].

[3] Al-Marzouqi, H. and AlRegib, G., "Using the coefficient of variation to improve the sparsity of seismic data," in *1st IEEE Global Conference on Signal and Information Processing*, 2013.

[4] Antonini, M., Barlaud, M., Mathieu, P., and Daubechies, I., "Image coding using wavelet transform," *Image Processing, IEEE Transactions on*, vol. 1, no. 2, pp. 205–220, 1992.

[5] Arivazhagan, S., Ganesan, L., and Kumar, T. S., "Texture classification using curvelet statistical and co-occurrence features," in *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, vol. 2, pp. 938–941, IEEE, 2006.

[6] Bajwa, W. U., Haupt, J. D., Raz, G. M., Wright, S. J., and Nowak, R. D., "Toeplitz-structured compressed sensing matrices," in *Statistical Signal Processing, 2007. SSP'07. IEEE/SP 14th Workshop on*, pp. 294–298, IEEE, 2007.

[7] Bianconi, F., González, E., Fernández, A., and Saetta, S. A., "Automatic classification of granite tiles through colour and texture features," *Expert Systems with Applications*, vol. 39, no. 12, pp. 11212–11218, 2012.

[8] Boashash, B., *Time frequency analysis*. Access Online via Elsevier, 2003.

[9] Buades, A., Coll, B., and Morel, J.-M., "A non-local algorithm for image denoising," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 2, pp. 60 – 65 vol. 2, june 2005.

[10] Buades, A., Coll, B., and Morel, J.-M., "Non-Local Means Denoising," *Image Processing On Line*, vol. 2011, 2011.

[11] Bulow, H. and Birk, A., "Spectral 6DOF registration of noisy 3D range data with partial overlap," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 35, pp. 954–969, April 2013.

[12] Burt, P. and Adelson, E., "The Laplacian pyramid as a compact image code," *Communications, IEEE Transactions on*, vol. 31, no. 4, pp. 532–540, 1983.

[13] Candes, E., Demanet, L., Donoho, D., and Ying, L., "Fast discrete curvelet transforms," *Multiscale Modeling & Simulation*, vol. 5, no. 3, pp. 861–899, 2006.

[14] Candes, E. and Donoho, D., "New tight frames of curvelets and optimal representations of objects with piecewise C^2 singularities," *Communications on pure and applied mathematics*, vol. 57, no. 2, pp. 219–266, 2003.

[15] Candès, E. J., Romberg, J., and Tao, T., "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *Information Theory, IEEE Transactions on*, vol. 52, no. 2, pp. 489–509, 2006.

[16] Chang, C.-L. and Girod, B., "Direction-adaptive discrete wavelet transform for image compression," *Image Processing, IEEE Transactions on*, vol. 16, pp. 1289–1302, May 2007.

[17] Choi, M., Kim, R., Nam, M., and Kim, H., "Fusion of multispectral and panchromatic satellite images using the curvelet transform," *Geoscience and remote sensing letters, IEEE*, vol. 2, no. 2, pp. 136–140, 2005.

[18] Cohen, L., *Time-frequency analysis*, vol. 778. Prentice Hall PTR New Jersey, 1995.

[19] Coifman, R. R. and Wickerhauser, M. V., "Entropy-based algorithms for best basis selection," *Information Theory, IEEE Transactions on*, vol. 38, no. 2, pp. 713–718, 1992.

[20] Conn, A. A. R., Scheinberg, K., and Vicente, L. N., *Introduction to derivative-free optimization*, vol. 8. SIAM, 2009.

[21] Dana, K., Van-Ginneken, B., Nayar, S., and Koenderink, J., "Reflectance and Texture of Real World Surfaces," *ACM Transactions on Graphics (TOG)*, vol. 18, pp. 1–34, Jan 1999.

[22] Daubechies, I. and others, *Ten lectures on wavelets*, vol. 61. SIAM, 1992.

[23] Daugman, J. G. and others, "Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters," *Optical Society of America, Journal, A: Optics and Image Science*, vol. 2, no. 7, pp. 1160–1169, 1985.

[24] De Valois, R. L., Albrecht, D. G., and Thorell, L. G., "Spatial frequency selectivity of cells in macaque visual cortex," *Vision research*, vol. 22, no. 5, pp. 545–559, 1982.

[25] Do, M. and Vetterli, M., "Wavelet-based texture retrieval using generalized gaussian density and Kullback-Leibler distance," *Image Processing, IEEE Transactions on*, vol. 11, pp. 146–158, Feb 2002.

[26] Do, M. and Vetterli, M., "The contourlet transform: an efficient directional multiresolution image representation," *Image Processing, IEEE Transactions on*, vol. 14, no. 12, pp. 2091–2106, 2005.

[27] Dong, Y. and Ma, J., "Wavelet-based image texture classification using local energy histograms," *Signal Processing Letters, IEEE*, vol. 18, pp. 247–250, April 2011.

[28] Fadili, M. and Starck, J.-L., "Sparse representation-based image deconvolution by iterative thresholding," *Astronomical Data Analysis ADA*, vol. 6, p. 18, 2006.

[29] Freeman, W. T. and Adelson, E. H., "The design and use of steerable filters," *IEEE Transactions on Pattern analysis and machine intelligence*, vol. 13, no. 9, pp. 891–906, 1991.

[30] Gómez, F. and Romero, E., "Rotation invariant texture characterization using a curvelet based descriptor," *Pattern Recognition Letters*, vol. 32, no. 16, pp. 2178–2186, 2011.

[31] Guo, K. and Labate, D., "Optimally sparse multidimensional representation using shearlets," *SIAM journal on mathematical analysis*, vol. 39, no. 1, pp. 298–318, 2007.

[32] Herrmann, F. J., "Randomized sampling and sparsity: getting more information from fewer samples," *Geophysics*, vol. 75, pp. WB173–WB187, 12 2010.

[33] Herrmann, F. and Hennenfent, G., "Non-parametric seismic data recovery with curvelet frames," *Geophysical Journal International*, vol. 173, no. 1, pp. 233–248, 2008.

[34] Hurley, N. and Rickard, S., "Comparing measures of sparsity," *Information Theory, IEEE Transactions on*, vol. 55, pp. 4723–4741, Oct 2009.

[35] Jacques, L., Duval, L., Chaux, C., and Peyr, G., "A panorama on multiscale geometric representations, intertwining spatial, directional and frequency selectivity," *Signal Processing*, vol. 91, no. 12, pp. 2699 – 2730, 2011. Advances in Multirate Filter Bank Structures and Multiscale Representations.

[36] Krommweh, J., "Tetrolet transform: A new adaptive haar wavelet algorithm for sparse image representation," *Journal of Visual Communication and Image Representation*, vol. 21, no. 4, pp. 364–374, 2010.

[37] Lasmar, N.-E. and Berthoumieu, Y., "Gaussian copula multivariate modeling for texture image retrieval using wavelet transforms," *Image Processing, IEEE Transactions on*, vol. 23, pp. 2246–2261, May 2014.

[38] Le Gall, D. and Tabatabai, A., "Sub-band coding of digital images using symmetric short kernel filters and arithmetic coding techniques," in *Acoustics, Speech, and Signal Processing, 1988. ICASSP-88., 1988 International Conference on*, pp. 761–764 vol.2, 1988.

[39] Lee, T. S., "Image representation using 2D Gabor wavelets," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 18, no. 10, pp. 959–971, 1996.

[40] Ma, J. and Plonka, G., "The curvelet transform," *Signal Processing Magazine, IEEE*, vol. 27, pp. 118 –133, March 2010.

[41] Mallat, S., *A wavelet tour of signal processing*. Academic press, 1999.

[42] Mallat, S. and Peyré, G., "A review of bandlet methods for geometrical image representation," *Numerical Algorithms*, vol. 44, no. 3, pp. 205–234, 2007.

[43] Mallat, S., "Geometrical grouplets," *Applied and Computational Harmonic Analysis*, vol. 26, no. 2, pp. 161 – 180, 2009.

[44] Mandal, T., Jonathan Wu, Q., and Yuan, Y., "Curvelet based face recognition via dimension reduction," *Signal Processing*, vol. 89, no. 12, pp. 2345–2353, 2009.

[45] Manjunath, B. S. and Ma, W.-Y., "Texture features for browsing and retrieval of image data," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 18, no. 8, pp. 837–842, 1996.

[46] Miri, M. and Mahloojifar, A., "Retinal image analysis using curvelet transform and multistructure elements morphology by reconstruction," *Biomedical Engineering, IEEE Transactions on*, vol. 58, no. 5, pp. 1183–1192, 2011.

[47] Murala, S., Maheshwari, R., and Balasubramanian, R., "Local tetra patterns: a new feature descriptor for content-based image retrieval," *Image Processing, IEEE Transactions on*, vol. 21, no. 5, pp. 2874–2886, 2012.

[48] Nelder, J. A. and Mead, R., "A simplex method for function minimization," *The computer journal*, vol. 7, no. 4, pp. 308–313, 1965.

[49] Ojala, T., Pietikäinen, M., and Harwood, D., "A comparative study of texture measures with classification based on featured distributions," *Pattern recognition*, vol. 29, no. 1, pp. 51–59, 1996.

[50] Plonka, G., Tenorth, S., and Rosca, D., "A new hybrid method for image approximation using the easy path wavelet transform," *Image Processing, IEEE Transactions on*, vol. 20, no. 2, pp. 372–381, 2011.

[51] Ponomarenko, N., Battisti, F., Egiazarian, K., Astola, J., and Lukin, V., "Metrics performance comparison for color image database," in *Fourth international workshop on video processing and quality metrics for consumer electronics*, vol. 27, 2009.

[52] Ro, Y. M., Kim, M., Kang, H. K., Manjunath, B., and Kim, J., "MPEG-7 homogeneous texture descriptor," *ETRI journal*, vol. 23, no. 2, pp. 41–51, 2001.

[53] Saevarsson, B., Sveinsson, J., and Benediktsson, J., "Combined wavelet and curvelet denoising of SAR images," in *Geoscience and Remote Sensing Symposium, 2004. IGARSS'04. Proceedings. 2004 IEEE International*, vol. 6, pp. 4235–4238, IEEE, 2004.

[54] Samaria, F. S. and Harter, A. C., "Parameterisation of a stochastic model for human face identification," in *Applications of Computer Vision, 1994., Proceedings of the Second IEEE Workshop on*, pp. 138–142, 1994.

[55] Selesnick, I. W., Baraniuk, R. G., and Kingsbury, N. C., "The dual-tree complex wavelet transform," *Signal Processing Magazine, IEEE*, vol. 22, no. 6, pp. 123–151, 2005.

[56] Simoncelli, E. P., Freeman, W. T., Adelson, E. H., and Heeger, D. J., "Shiftable multiscale transforms," *Information Theory, IEEE Transactions on*, vol. 38, no. 2, pp. 587–607, 1992.

[57] Simoncelli, E., Freeman, W., Adelson, E., and Heeger, D., "Shiftable multiscale transforms," *Information Theory, IEEE Transactions on*, vol. 38, pp. 587–607, March 1992.

[58] Starck, J.-L., Fadili, J., and Murtagh, F., "The undecimated wavelet decomposition and its reconstruction," *Image Processing, IEEE Transactions on*, vol. 16, no. 2, pp. 297–309, 2007.

[59] Starck, J., Candès, E., and Donoho, D., "The curvelet transform for image denoising," *Image Processing, IEEE Transactions on*, vol. 11, no. 6, pp. 670–684, 2002.

[60] Sumana, I., Islam, M., Zhang, D., and Lu, G., "Content based image retrieval using curvelet transform," in *Multimedia Signal Processing, 2008 IEEE 10th Workshop on*, pp. 11 –16, Oct. 2008.

[61] Ulfarsson, M., Sveinsson, J., and Benediktsson, J., "Speckle reduction of SAR images in the curvelet domain," in *Geoscience and Remote Sensing Symposium, 2002. IGARSS'02. 2002 IEEE International*, vol. 1, pp. 315–317, IEEE, 2002.

[62] van den Berg, E. and Friedlander, M., "SPGL1: A solver for large-scale sparse reconstruction," *Online: http://www. cs. ubc. ca/labs/scl/spgl1*, 2007.

[63] Wang, Z., Bovik, A., Sheikh, H., and Simoncelli, E., "Image quality assessment: From error visibility to structural similarity," *Image Processing, IEEE Transactions on*, vol. 13, no. 4, pp. 600–612, 2004.

[64] Wang, Z.-Z. and Yong, J., "Texture analysis and classification with linear regression model based on wavelet transform," *Image Processing, IEEE Transactions on*, vol. 17, pp. 1421–1430, Aug 2008.

[65] Zhang, D., Islam, M. M., Lu, G., and Sumana, I. J., "Rotation invariant curvelet features for region based image retrieval," *International journal of computer vision*, vol. 98, no. 2, pp. 187–201, 2012.

[66] Zhihua-Xie, Liu, G., Wu, S., and Lu, Y., "A fast infrared face recognition system using curvelet transformation," in *Electronic Commerce and Security, 2009. ISECS '09. Second International Symposium on*, vol. 2, pp. 145–149, 2009.

[67] Zujovic, J., Pappas, T. N., and Neuhoff, D. L., "Structural texture similarity metrics for image analysis and retrieval," *Image Processing, IEEE Transactions on*, vol. 22, no. 7, pp. 2545–2558, 2013.

Curvelet Transform with Adaptive Tiling

Hasan Al-Marzouqi

86 Pages

Directed by Professor Ghassan AlRegib

In this dissertation, we address the problem of adapting frequency domain tiling using the curvelet transform as the basis algorithm. The optimal tiling, for a given class of images, is computed using denoising performance as the cost function. The major adaptations considered are: the number of scale decompositions, angular decompositions per scale/quadrant, and scale locations. A global optimization algorithm combining the three adaptations is proposed. Denoising performance of adaptive curvelets is tested on seismic and face data sets. The developed adaptation procedure is applied to a number of different application areas. Adaptive curvelets are used to solve the problem of sparse data recovery from subsampled measurements. Performance comparison with default curvelets demonstrates the effectiveness of the adaptation scheme. Adaptive curvelets are also used in the development of a novel image similarity index. The developed measure succeeds in retrieving correct matches from a variety of textured materials. Furthermore, we present an algorithm for classifying different types of seismic activities.